ITHEA

International Journal INFORMATION MODELS & ANALYSES

2015 Volume 4 Number 1

International Journal
INFORMATION MODELS & ANALYSES
Volume 4 / 2015, Number 1

IJ IMA is official publisher of the scientific papers of the members of the ITHEA® International Scientific Society

IJ IMA rules for preparing the manuscripts are compulsory.
The **rules for the papers** for ITHEA International Journals are given on www.ithea.org .
The camera-ready copy of the paper should be received by ITHEA® Submission system http://ij.ithea.org .
Responsibility for papers published in IJ IMA belongs to authors.

International Journal "INFORMATION MODELS AND ANALYSES" Volume 4, Number 1, 2015

Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria, in collaboration with
Institute of Mathematics and Informatics, BAS, Bulgaria,
V.M.Glushkov Institute of Cybernetics of NAS, Ukraine,
Universidad Politechnika de Madrid, Spain,
Hasselt University, Belgium
Institute of Informatics Problems of the RAS, Russia,
St. Petersburg Institute of Informatics, RAS, Russia
Institute for Informatics and Automation Problems, NAS of the Republic of Armenia,

Publisher: **ITHEA®**

Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org, e-mail: info@foibg.com
Technical editor: **Ina Markova**

Printed in Bulgaria

ISSN 1314-6416 (printed) **ISSN 1314-6432 (Online)**

A STUDY OF INTELLIGENT TECHNIQUES FOR PROTEIN SECONDARY STRUCTURE PREDICTION

Hanan Hendy, Wael Khalifa, Mohamed Roushdy, Abdel Badeeh Salem

Abstract: *Protein secondary structure prediction has been and will continue to be a rich research field. This is because the protein structure and shape directly affect protein behavior. Moreover, the number of known secondary and tertiary structures versus primary structures is relatively small. Although the secondary prediction started in the seventies but it has been together with the tertiary structure prediction a topic that is always under research. This paper presents a technical study on recent methods used for secondary structure prediction using amino acid sequence. The methods are studied along with their accuracy levels. The most known methods like Neural Networks and Support Vector Machines are shown and other techniques as well. The paper shows different approaches for predicting the protein structures that showed different accuracies that ranged from 50% to over than 90%. The most commonly used technique is Neural Networks. However, Case Based Reasoning and Mixed Integer Linear Optimization showed the best accuracy among the machine learning techniques and provided accuracy of approximately 83%.*

Keywords: *Bioinformatics, Machine Learning, Protein Secondary Structure Prediction.*

ACM Classification Keywords: *I.2 Artificial Intelligence, H.4 Information System Applications, H.4.2 Types of systems decision support*

Introduction

Protein structure prediction is known as predicting -getting- the secondary and/or tertiary structure from linear amino acid sequence known as Primary structure. Predicting the secondary structure of proteins helps in many domains. Some of these domains can be: knowing the functionally of the protein, drug design, the design of novel enzymes and disease detection such as “Alzheimer’s” and other diseases related to cancer [Camacho et al, 2012] and much more. Moreover, predicting the secondary structure is a basic and crucial step in the tertiary structure prediction. Tertiary structures that are known are relatively very small. In mid-2011, there were only 70,000 known tertiary structures in the PDB –Protein Data Bank- compared to 12.5 million protein sequences in the RefSeq database [Kister, 2013], so it’s very difficult to keep track of secondary and tertiary structures in the same pace of primary structures detection.

In this paper, section one presents a short biological background showing the important terminologies that are used all through the paper. Then, in section two the prediction methodologies are presented as a sequence of methodologies/ techniques. Each method is presented along with its accuracy and a brief description of the method. Methods presented can be categorized as statistical/probabilistic ones [Chou–Fasman, 2014; Garnier et al, 1996], Neural Networks which is the most common techniques used [Chandonia, 1995; Silva, 2005; Rost, 1996] and Case Based Reasoning [Glasgow et al, 2006]. Finally, the current research trends used in secondary structure prediction are presented that uses Support Vector Machines [Sui et al, 2011] and Swarm Intelligence (Bee Colony) [Li, 2014]. Finally discussing mixing more than one predictor as it is thought to be the future trend of secondary structure prediction [Wei, 2011].

Biological Background

Proteins are known to be large biological molecules built up from one or more Amino Acid residues. Proteins are responsible for many vital functions in the human body, for example: replicating DNA, responding to stimuli, metabolic functions and a lot more. There exists twenty amino acids –each has a unique shape and prefix letter – which builds up any protein. Amino acids are formed from Oxygen, Nitrogen, Hydrogen and Sulfur atoms [Protein, 2014]. The standard amino acids are shown in Figure 1.

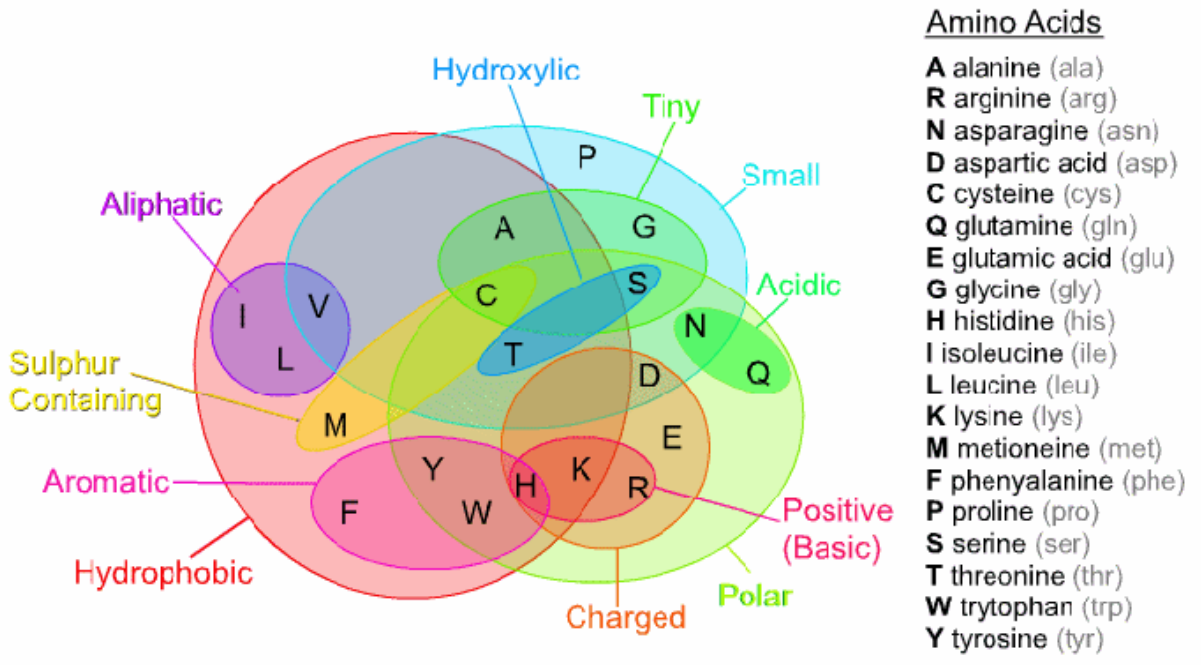


Figure 1. Venn diagram of boundaries that symbolizes the universal set of 20 common amino acids [Esquivel, 2013]

When amino acids interact together in order to be able to perform their functionalities, the result is called Structure. To understand the functions of proteins at a molecular level, it is often necessary to determine their three-dimensional structure. Protein structure can be organized into four distinct levels as shown in Figure 2.

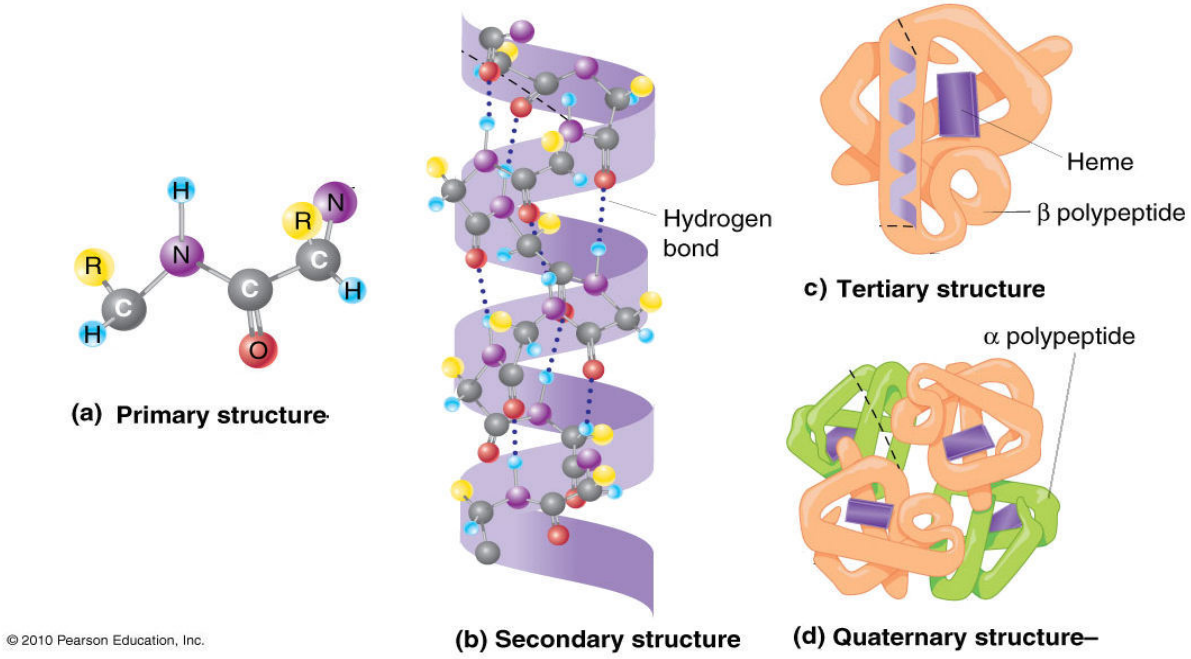


Figure 2. Four levels of Protein Structure [Russell, 2009]

The primary structure, which is the basic structure of Protein describes the linear sequence of amino acid in polypeptide chain. The primary structure is always noted to by one or three letters. Secondary structure is formed when amino acids interact together forming hydrogen bonds. According to the DSSP – Dictionary of Protein Secondary Structures – [DSSP, 2014] secondary structures can be seen as eight classes namely: H (alpha helix), G(helix-3), I (Hilex-5), E(stand), B(beta bridge), T(Turn), S(bend), -(irregular). These structures are often mapped to three levels: alpha helix (H) and beta strands/sheets (E) and coil (C) which covers S, T and - states.

The three dimensional structure –also known as Tertiary structure- is formed when alpha helix and/or beta sheet interact together forming a more complex geometrical shape forming beta-peptide. Quaternary is stabilized by the same non-covalent interactions and disulfide bonds of the tertiary structure.

Secondary Structure Prediction Techniques/Algorithms

Protein secondary structure prediction is defined as the set of techniques and algorithms in bioinformatics that aim to predict the local secondary structures based only on the knowledge of their primary structure. As stated by Sara Silva [Silva, 2005], secondary structure predication passed through generations. These generations differ from one another by the techniques used and the prior knowledge of protein, starting by pure statistical methods getting into machine learning and intelligent techniques.

A. Statistical Generation

- This generation is characterized that all its methods are based on statistical analysis of single residue. The first probabilistic method that is considered the starting point of secondary structure prediction is “Chou-Fasman Method” [Chou–Fasman, 2014];
- The method is based on analysis of the relative frequencies of each amino acid in alpha helices, beta sheets, and turns based on known protein structures solved with X-ray crystallography. From these frequencies a set of probability parameters were derived for the appearance of each amino acid in each secondary structure type, and these parameters are used to predict the probability that a given sequence of amino acids would form a helix, a beta strand, or a turn in a protein.

The second most important method in this generation is the GOR method (Garnier-Osguthorpe-Robson) [Garnier et al, 1996]. The GOR method takes into account not only the propensities of individual amino acids to form particular secondary structures, but also the conditional probability of the amino acid to form a secondary structure given that its immediate neighbors have already formed that structure. The method is therefore essentially Bayesian in its analysis.

B. Enhanced Statistical Generation

This set of techniques introduced the usage of local interactions along with segment statists in the prediction approach. The start was GOR III which was an improvement to GOR. It was the first to use local interactions between amino acids. This means that, to predict the secondary structure of a given amino acid, the information about which amino acids are following and preceding it in the sequence is used [Garnier et al, 1996; Silva, 2005].

C. Machine Learning Generation

These set of techniques are homology based, studying the local interactions and alignments. Also introducing intelligent techniques such as neural networks, case based reasoning and much more.

– Neural Networks:

Qian and Terrence [Qian et al, 1988] introduced one of the first Neural Networks used for secondary structure prediction. They worked on a network with 17 input groups having 21 units per group, 40 hidden units and three output units. The usage of Neural Networks then started evolving and different architectures were studied targeting better prediction accuracy.

Chandonia et al [Chandonia, 1995], used the standard amino acid sequence as the input to their Neural Network, then they used the output of this prediction along with other information to help predict the structural class (tertiary structure). At last they used the structural class predicted in a third network to predict again the secondary structure to reach a better accuracy.

Later on, The Profile neural network prediction from HeiDelber (PHD) [Rost, 1996] methodology was introduced. It is considered the backbone of all proceeding methods. The PHD has four processing levels the output of each level is used as input for the following. It starts with a level that has the amino acid sequence as the input and outputs the likelihood that it belongs to alpha-helix, beta-strands or others (loop). Then the second level, uses this likelihood with some global information about the protein (for example, its length) to calculate a new likelihood. The third level chooses the classification of protein. Finally, filters the result observing common errors and unreasonable results.

More advanced neural networks were then proposed by Pollastri et al [Pollastri et al, 2002]. They used bidirectional recurrent neural network. Also they introduced the Second version of the SSpro program for secondary structure prediction.

– Case Based Reasoning:

Another technique different than Neural Networks was introduced by Janice et al [Glasgow et al, 2006]. They used Case Based Reasoning technique to predict the secondary structure of protein. They present the protein by a 2D map then they use case matching to query the cases that have common features with the new case. Getting these cases they are capable of getting the structure of protein.

– Swarm Intelligence:

Swarm intelligence is also used in Protein secondary structure prediction. Bai Li et at [Li, 2014] introduced the use of Artificial Bee Colony (ABC) algorithm. They used internal feedback strategy based ABC. It was proved to be effective to improve convergence rate also it was stated that this approach is better in exploration than exploitation.

– Support Vector Machine:

Haifeng Sui et al [Sui et al, 2011] proposed that Hybrid SVM can enhance the prediction accuracy of protein secondary structure. They proposed that combining physicochemical properties of amino acid residues with position-specific scoring matrices containing evolutionary information. The accuracy of this approach was not clearly proved but it was stated that it's better that it has a better ability than other methodologies.

– Combined Methods:

A combined method was introduced by Y. Wei et al [Wei, 2011]. They combined seven secondary structure prediction methods. The prediction is accomplished using the value from each predictor these values are then combined to find out the likelihood of the amino acid sequence.

Another combined method was proposed by Camacho, R. et al [Camacho et al, 2012]. This method reached an accuracy of almost 84.9% (in the prediction of α -helices) and 99.6% (in the prediction of the inner points of β -strands). This method combined rule induction algorithms, decision trees, functional trees, Bayesian methods and other algorithms.

Other methodologies that aim to enhance the prediction accuracy are introduced as well. Some methodologies uses the prediction of the tertiary structure as input to the prediction step as shown in [Chandonia, 1995]. Others combine the result from 3 predictors. The aim is always to increase the confidence level of predicting the amino acid sequence to be alpha Helix (H), Beta strands [E] or other (Loop). The three prediction always occurs to be one of three possibilities: 3:0 which indicates that the three methods predicted the sequence the same, 2:1 which leads to majority decision, or 1:1:1 which indicates a tie in which each predictor had a different output and in this case the amino acid sequence is predicted to be L state. The first two one of the states is dominant and it is chosen to represent the sequence [Albrecht et al, 2003].

The main observations from Table 1:

1- Protein secondary structure prediction started by statistical methods at which the prediction accuracy was very low. Then the accuracy started to increase when intelligent techniques arose. Getting into a fairly better accuracy when combining more than one methodology.
2- Neural Network with its variations is the most commonly used approach for Protein secondary structure prediction.
3- Other intelligent techniques are not yet mature as Neural Networks, although they have better accuracy. The approaches that tend to have accuracy better than 90% are those which use mixed predictors. Also SVM showed 90% accuracy only for β –strands prediction.

Table 1. Comparison of Secondary Structure Prediction Methodologies

Authors	Method / Algorithm	Dataset	Accuracy
Statistical and Enhanced Statistical Techniques			
Peter Y. Chou and Gerald D. Fa sman [Chou–Fasman, 2014]	Chou-Fasman method	-	50–60%
Jean Garnier et al [Garnier et al, 1996]	GOR & GOR III	Database of 267 protein structures	60%
Machine Learning Techniques			
Qian and Sejnowski [Qian et al, 1988]	Neural Network with window size 13	106 proteins	64.3%
Chandonia and Karplus [Chandonia, 1995]	Neural Networks	Set of 62 globular proteins (69 chains)	Secondary structure prediction 62.64% Class prediction 73.9%
Rost [Silva, 2005; Rost, 1996]	PHD	-	better than 72% about 74% of the segments are correctly predicted
Gianluca et al [Pollastri et al, 2002]	Recurrent Neural Networks and Profiles	Four data sets TRAIN for training and R126, EVA, and CASP4 for testing.*	78%
Janice et al [Glasgow et al, 2006]	Case Based Reasoning	-	83%
Y. Wei et al	Mixed integer linear	3000 proteins are selected from PDB as	83.04%

Authors	Method / Algorithm	Dataset	Accuracy
Statistical and Enhanced Statistical Techniques			
[Wei, 2011]	optimization	the training set.	
Camacho,R. et al [Camacho et al, 2012]	Machine Learning (rule induction, decision trees, functional trees, Bayesian methods)	1499 protein structures from the PDB	84.9% (in the prediction of α -helices) and 99.6% for β -strands
Haifeng Sui et al [Sui et al, 2011]	HSVM	462 proteins from the CB513 for training 3 for testing RS126, CB513 and CASP9*	independent predictions for more than 55% of all amino acid residues with accuracies of up to 90%
Bai Li et at [Li, 2014]	Bee Colony	-	-

* EVA, CASP4, CASP9, CB513, R126 and RS126 are all databases of protein structures.

Conclusion and Future Work

Sequence based prediction enjoys strong interest and finds its applications in various fields. Although it started long ago with probabilistic methods, recent research tries to find suitable intelligent techniques to enhance the prediction accuracy. Finding better methodologies to predict the secondary structure helps not only in the secondary structure domain but also in the tertiary structure domains.

We showed in this paper the three different generations of protein secondary structure prediction, namely the statistical generation, Enhanced statistical generation and Machine learning. We have demonstrated some of the most used techniques in each generation. Having an objective comparison among prediction methods is very difficult and not relevant in all cases. As shown each method used a different dataset for testing, also different definition for the input sequence and topology (some used variant length while others not).

However, the highest accuracies reached are from Case Based Reasoning approach which generated an accuracy of 83% and Mixed Integer Linear Optimization generated an accuracy of 83.4%. Our future work will go towards using more than one predictor and combine their results to reach a better accuracy and confidence level.

Bibliography

[Albrecht et al, 2003] M. Albrecht, Silvio C.E. Tosatto, Thomas Lengauer and Giorgio Valle, "Simple consensus procedures are effective and sufficient in secondary structure prediction," peds -Protein Engineering design & Selection-, vol. 16, no. 7, pp. 459-462, 2003. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[Camacho et al, 2012] R. Camacho, R. Ferreira, N. Rosa, V. Guimarães, N. A. Fonseca, V. S. Costa, M. D. Sousa and A. Magalhães, "Predicting the secondary structure of proteins using Machine Learning algorithms," Int. J. Data Mining and Bioinformatics, vol. 6, no. 6, pp. 571-584, 2012.

[Chandonia, 1995] John-Marc Chandonia and Martin Karplus, "Neural networks for secondary structure and structural class predictions," Protein Science, vol. 4, no. 2, p. 275–285, February 1995.

[Chou–Fasman, 2014] "Chou–Fasman method," Available: http://en.wikipedia.org/wiki/Chou%E2%80%93Fasman_method, [Accessed November 2014].

[DSSP, 2014] "DSSP", Available: http://www.cmbi.ru.nl/dssp.html, [Accessed November 2014].

[Esquivel, 2013] Rodolfo O. Esquivel, Moyocoyani Molina-Espíritu, Frank Salas, Catalina Soriano, Carolina Barrientos, Jesús S. Dehesa and José A. Dobado, "Decoding the Building Blocks of Life from the Perspective of Quantum Information," Advances in Quantum Mechanics, pp. 641-669, 2013.

[Garnier et al, 1996] Jean Garnier, Jean-François Gibrat and Barry Robson, "[32] GOR method for predicting protein secondary structure from amino acid sequence," Methods in Enzymology, vol. 266, p. 540–553, 1996.

[Glasgow et al, 2006] J. Glasgow, Tony Kuo and Jim Davies, "Protein Structure from Contact Maps: A Case-Based Reasoning Approach," Information Systems Frontiers, vol. 8, no. 1, pp. 29-36, February 2006.

[Kister, 2013] A. E. Kister, Protein Supersecondary Structures, Humana Press, 2013.

[Li, 2014] Bai Li, Ya Li and Ligang Gong, "Protein secondary structure optimization using an improved artificial bee colony algorithm based on AB off-lattice model," Engineering Applications of Artificial Intelligence, vol. 27, p. 70–79, 2014.

[Pollastri et al, 2002] Gianluca Pollastri, Darisz Przybylski, Burkhard Rost and Pierre Baldi, "Improving the Prediction of Protein Secondary Structure in Three and Eight Classes Using Recurrent Neural Networks and Profiles," PubMed, pp. 228-235, May 2002.

[Protein, 2014] "Protein", Available: http://en.wikipedia.org/wiki/Protein, [Accessed November 2014].

[Qian et al, 1988] Ning Qian and T. J. Sejnowski, "Predicting the secondary structure of globular proteins using neural network models," Journal of Molecular Biology, vol. 202, no. 4, p. 865–884, August 1988.

[Rost, 1996] B. Rost, "[31] PHD: Predicting one-dimensional protein structure by profile-based neural networks," Methods in Enzymology, vol. 266, p. 525–539, 1996.

[Russell, 2009] P. J. Russell, iGenetics: A Molecular Approach, Third ed., PEARSON, 2009.

[Silva, 2005] S. Silva, "Predicting Protein Secondary Structure using Artificial Neural Networks - A Short Tutorial," 2005

[Sui et al, 2011] Haifeng Sui, Wu Qu, Bingru Yan and LiJun Wang, "Improved Protein Secondary Structure Prediction Using a Intelligent HSVM Method with a New Encoding Scheme," International Journal of Advancements in Computing Technology, vol. 3, no. 3, pp. 239-250, April 2011.

[Wei, 2011] Y. Wei, J. Thompson and C. A. Floudas, "CONCORD: a consensus method for protein secondary structure prediction via mixed integer linear optimization," Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science, pp. 831-850, November 2011.

Authors' Information

Hanan Hendy *– Teaching Assistant at Computer Science Department, Faculty of Computer and Information Sciences, Ain Shams University; e-mail: hanan.hendy@cis.asu.edu.eg*

Major Fields of Scientific Research: Bioinformatics, Artificial Intelligence

Wael Khalifa *– Lecturer at Computer Science Department, Faculty of Computer and Information Sciences, Ain Shams University; e-mail: wael.khalifa@cis.asu.edu.eg*

Major Fields of Scientific Research: Biometrics, Bio and Medical Informatics

Mohamed Roushdy *– Professor at Computer Science Department and Dean of Faculty of Computer and Information Sciences, Ain Shams University; e-mail: mroushdy@cis.asu.edu.eg*

Major Fields of Scientific Research: Artificial Intelligence, Medical Expert Systems

Abdel Badeeh Salem *– Professor at Computer Science Department, Faculty of Computer and Information Sciences, Ain Shams University; e-mail: absalem@cis.asu.edu.eg*

Major Fields of Scientific Research: Knowledge Engineering, Artificial Intelligence, Biomedical Informatics

DEVELOPMENT AND ANALYSIS OF GENETIC ALGORITHM FOR TIME SERIES FORECASTING PROBLEM

Leonid Hulianytskyi, Anna Pavlenko

Abstract: *This paper presents developed genetic-based algorithm for time series forecasting problem and describes approaches to learning procedures design. Different techniques of population representation, recombination, formation of niches, calculation of fitness, conflict resolution methods are proposed. Results of computational experiments with real time series are analyzed.*

Keywords: *forecasting, genetic-based machine learning, rule-based forecasting, genetic algorithm, time series forecasting, evolutionary algorithms.*

ACM Classification Keywords: *I.2.8 Problem Solving, Control Methods, and Search*

Introduction

Applications of evolutionary computation to machine learning are referred to as genetic-based machine-learning (GBML). Evolutionary computation (EC) techniques belong to the class of optimization tools, inspired by biological processes. The main idea of EC lies in the iterative modification of the population of individuals (candidate solutions of the problem – chromosomes) with selection and recombination procedures.

Rule-based genetic algorithms (GA) are successfully applied to the solution of machine learning problems due to natural scalability, parallelization, noise resilience, flexibility of objective function, universality of computational scheme and possibility of using heuristics for data representations [Kovacs, 2010]. On the other hand, rule-based forecasting is able to take into account several time series at once and consider existing causal relationships in complex economic processes, which are significantly affected by various factors.

We consider time-series forecasting as a special type of *classification* – or supervised learning task (learning algorithm knows about class of an example). Classification problem can be defined as follows: Given a set of instances $\Xi = \{i_1,...,i_n\}$, each of them labeled with a finite set of classes $C = \{c_1,...,c_m\}$, one wants to create a certain theory T based on Ξ [Bacardit, 2004].

Learning algorithm produces instances, where each one contains a finite and fixed set of elements – attributes, that represent features of the instances. This research operates with nominal attributes – values can be obtained from given discrete set. Set of classification rules is a result of the learning

algorithm. Classification rules are usually represented as follows: **If** *condition* **Then** *action.* The left hand side of each rule (*Condition*) is a conjunction of one or more tests involving feature values. The right hand side of a rule (*Action*) indicates the class is assigned to examples, which match its left hand side. Classification rules in the set might overlap [De Jong K.A., 1993]. GBML systems use sets of rules as knowledge representation.

This paper proposes a forecasting method based on GA [Mahfoud, 1996] and explores issues of its implementation and application.

Objectives of the research include proposing an approach to GA application for constructing rule base that should be able to recognize instances of the target concept correctly and discriminate them from objects that do not belong to it. This paper also describes approach to fitness-function calculation, data representation and conflict resolving scheme for forecasting problem.

General Scheme of the Genetic Algorithm

The underlying commonality of GBML is the use of an EC as the search mechanism.

Rule-based forecasting process is divided into the following steps:

1. Learning phase – construction of the rule base that describes all dependencies in time series using selected learning algorithm (GA in this research). Current algorithm employs incremental learning scheme – knowledge base could be updated as new examples arrive to the system.

2. In the match phase, all rules are checked for admissibility of use at the particular position, and conflicting rules set is formed.

3. In the conflict resolution phase, we select rules that would be used for forecasting from the set of conflicting rules.

4. During action phase, we apply rules to the forecast for the predefined period.

GA is a common technique and has different implementations for each problem, but it can be generally defined with scheme represented in the figure 1 [Eiben A.E., 2007]. GA works with population of individuals – admissible solutions to the problem. Initial population can be composed randomly, or it can be filled with expert solutions. After initiation, each individual is evaluated via fitness function. When population is fully constructed, reproduction cycle begins: the population is being modified through parents' selection, recombination, probabilistic mutation and evaluation. It is worth to mention that many operations in GAs are stochastic. Current algorithm is steady-state – only a subset of individuals (usually, only two of them) is modified during each iteration [Fernandez A., 2010].

There are two main approaches for representing a rule set in terms of GA. In the Pittsburgh approach, one chromosome encodes one rule, therefore, a problem of complexity of structures appear. In the

Michigan approach, one rule is usually represented by many chromosomes, therefore, credit assignment appears to be more difficult.

In terms of GA, a single rule is a chromosome and rule base is a population. Population is a set of possible solutions. It also could be defined as a single unit of evolution. Quantity of different solutions in the population represents its *diversity*.

```
BEGIN
  INITIALISE population with random candidate solutions;
  EVALUATE each candidate;
  REPEAT UNTIL ( TERMINATION CONDITION is satisfied ) DO
    1 SELECT parents;
    2 RECOMBINE pairs of parents;
    3 MUTATE the resulting offspring;
    4 EVALUATE new candidates;
    5 SELECT individuals for the next generation;
  OD
END
```

Figure 1. Pseudo-code of GA general scheme

Conflict resolution strategy is used in rule-based systems to make decision which rule should be applied if there are several applicable ones. There are many different approaches for conflict resolution [Sasikumar M., 2007]. One of the easiest is rule ordering (first come, first served), or selecting a rule that appeared in the conflict set first. Another way to resolve the conflict is to select the rule that is the most specific, i.e. the rule with the longest conditional part is preferred. According to the recency strategy, the rule, that uses the most recently added data, is applied. Priority strategy selects rules that are important in some ways (usually, priorities are determined by expert). Refractoriness approach ignores rules, which had already been selected (rules are removed from the conflict set after selection). A combination of different strategies is often used to solve forecasting problems. Sensitivity and stability throughout the system depends on effectiveness of the conflict resolution schemes.

Rule Representation

In this paper, population is represented using Pittsburgh approach, i.e. *rule* is an *if-then* construction and consists of arbitrary number of conditions w (atomic subrules) and final value w_c. Atomic subrule is the proposition that growth of time series values in random points belongs to some interval retrieved by time-series ranking [Гуляницкий Л.Ф., 2014]:

$$w = (F_{t_1} - F_{t_2}) \in [y_b, y_{b+1})$$

where F_t – value of the time series at the point t;

t_1, t_2 – arbitrary points,

y_b, y_{b+1} – real numbers, used for increment ranking ,

$b = \overline{-m, m-1}$, $m \in N$.

Here:

$$y_b = \frac{F_{\max}}{m} \times b$$

where $F_{\max}$ – maximal absolute value of the forecast (positive or negative), which is expertly determined for particular model. Ranking procedure results in construction of $2m$ intervals (for negative and positive gains). It is worth noting that the prediction algorithm, based on GAs, is able to work with multiple time series and, therefore, take into account causal factors of complex systems. In this case, parameters $F_{\max}$ and m must be set individually for each series [Гуляницкий Л. Ф., 2014а].

Full rule is defined as:

$$\Omega_j = if\left(\bigwedge_{i=1}^{z} w_i\right) \quad then \quad w_c,$$

where w_i – subrule of the conditional part,

i – index of the subrule,

$i = \overline{1, z}$, z – rule length (number of subrules in the full rule),

w_c – subrule, which is applied in case of triggering condition *if*.

Note that the rules use relative indexing instead of absolute values, i.e. all subrule indices are determined by the distance to the initial zero index.

Thus, all rules represent pattern that is applied to the time series via serial shift:

$$\tilde{w} = \Delta(s, k) \in [y_b, y_{b+1}),$$

where s, k – indices that define the distance relative to the initial index of rules.

Consider a graphical interpretation of a rule with one subrule and its application to a time series (Figure 2). The rule $if(\Delta(s_0, s_7) \in [0,50))$ $then$ $(\Delta(s_7, s_{16}) \in [0,50))$ indicates that if the growth function of two points (the distance between which is 7) is in the range from 0 to 50, then the growth between the 7th and the 16th points is in the range from 250 to 300.

Pattern (rule) is verified for all t. It is obvious that the rule is satisfied for $s_0 = 10$, $s_1 = 17$, $s_2 = 26$.

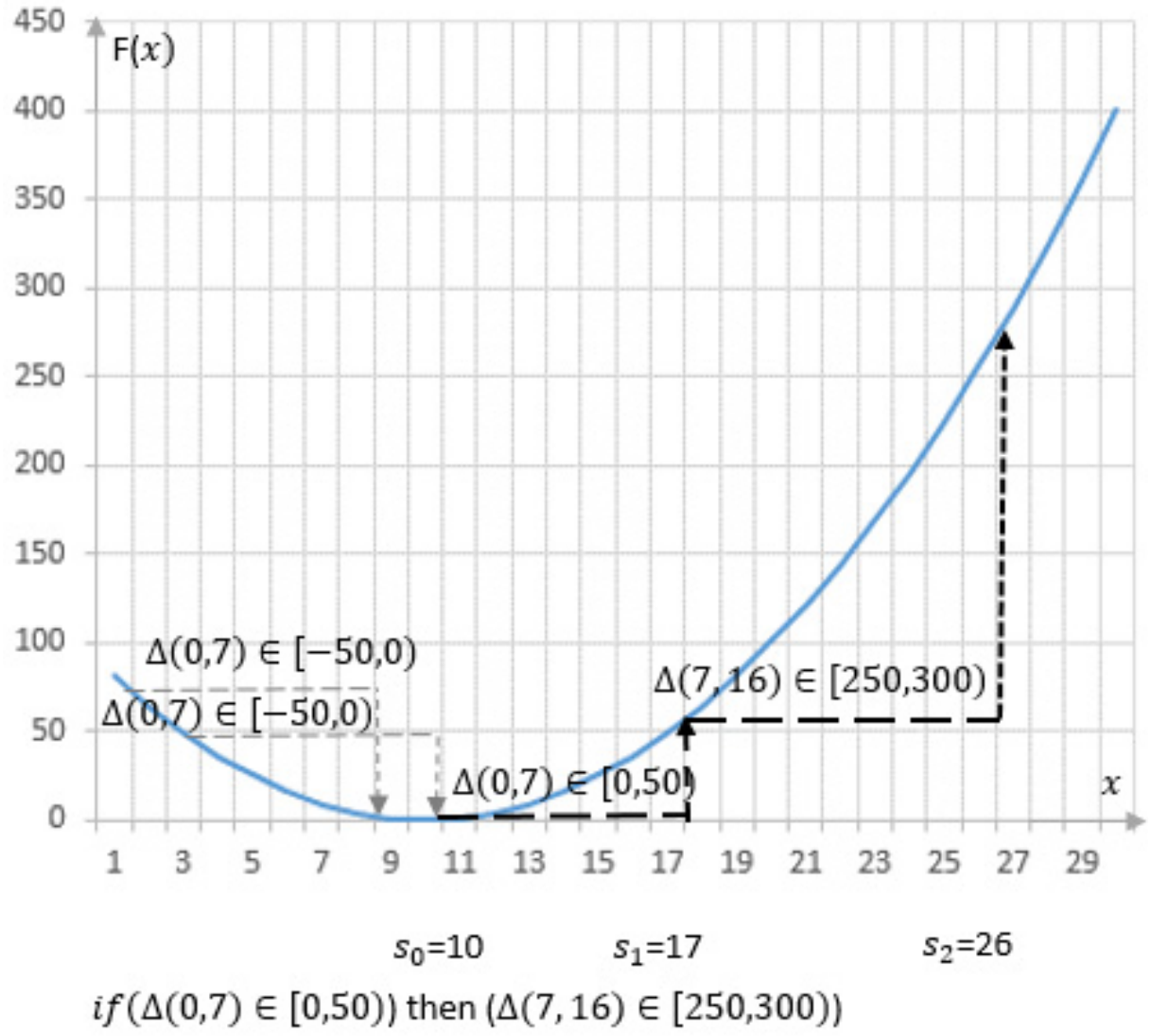


Figure 2. Graphical interpretation of the rule application to the time series

Fitness Function

The quality of each individual of the population is evaluated with fitness function ϕ, that takes into account frequency φ of the rule in the time series F and length of the rule z:

$$\phi = \begin{cases} \alpha z + (1-\alpha)\varphi, & \varphi > \delta \\ 0, & \varphi \le \delta \end{cases},$$

where δ – parameter of the algorithm that controls minimal allowed frequency of the rule,

α – parameter of the algorithm that adjusts weight values for frequency and rule length.

Despite definition of the rule as a conjunction of conditions, we assume that the rule is satisfied at the point t, if the share of satisfied subrules ω_i is higher than a defined parameter ϖ, and ω_c is completely satisfied.

Rule satisfaction frequency $\varphi(\Omega_j)$ is calculated by sequential shifting of rule indices along time series and reviewing rule compliance at each point t:

$$\varphi(\Omega_j) = \sum_{t=1}^{N} \Omega_j^t .$$

where $\Omega_j{}^t$ indicates whether rule is satisfied ($\Omega_j{}^t = 1$, if rule is satisfied at the moment t, $\Omega_j{}^t = 0$ in other cases).

Rule similarity

When GAs are used for optimization, the goal is generally to return a single value – the best solution found. Usually, if the conventional GA performs a sufficient number of iterations, the entire population converges to the neighborhood of one solution [Mahfoud SW, 1995]. Such approach is not suitable for forecasting problem, since it is necessary to obtain a sufficiently broad base of rules to make a final forecast.

GAs, that use niching procedures, are able to find and use different rules from the same population. The basic idea is simultaneous optimization in several areas of the search space, performed by reducing competition between sufficiently dissimilar individuals. In financial forecasting, different rules within a single population of GA are able to make predictions for various market and company conditions [Sam Mahfoud, 1996].

There is a variety of niching strategies: *fitness sharing* [Goldberg DE., 1987] – reducing the value of fitness for similar individuals in a population; *crowding* – new individuals replace older elements of the population; *sequential niching* – forcing restart of traditional GA in different search spaces and so on.

To implement niching procedure for the genetic-based forecasting algorithm we define the concept of the rules similarity as the degree of similarity of chromosomes genotypes. Two rules are similar, if the percentage $\tilde{w}$ of their similar subrules is higher than specified model parameter σ, $\sigma = \overline{1,\ldots,100}$.

Subrules $w_1 = \Delta(s_1, k_1) \in [y_{b_1}, y_{b_1+1})$ and $w_2 = \Delta(s_2, k_2) \in [y_{b_2}, y_{b_2+1})$ are similar, if their subrules have similar shift with respect to the initial rule index, cover segment of similar size on the time series and belong to the similar range of values:

$$|s_1 - s_2| \le \vartheta, \quad \vartheta = \overline{1,n}$$

$$|(s_1 - k_1) - (s_2 - k_2)| \le \varsigma, \quad \varsigma = \overline{1,n},$$

$$|b_1 - b_2| \le \gamma, \quad \gamma = \overline{1,n}$$

$$s_1 > k_1, \quad s_2 > k_2, \quad s_1 = \overline{1,n}, \quad k_1 = \overline{1,n}, \quad s_2 = \overline{1,n}, \quad k_2 = \overline{1,n}$$

where s_1, s_2 – starting subrule indices,

k_1, k_2 – ending subrule indices,

ϑ – model parameter that defines permissible shifting error of the starting index of similar subrules,

ς – model parameter that defines permissible error of segment length,

b_1, b_2 – indices that define intervals (ranks) for subrules;

γ – model parameter that defines permissible rank error of similar subrules.

Two rules are similar, if the percentage of similar subrules is higher than a defined model parameter ρ.

The procedure for similarity search is used to eliminate redundant rules at the recombination phase, at the stage of survival selection, at selection for recombination in inbreeding and outbreeding methods etc.

Initialization

Initial population has a fixed size, specified by the parameter of the algorithm. It is constructed from a given time series, considering the minimal allowed size of rules, specified by the parameter.

Subrules are constructed by selecting random points of the time series and comparing the corresponding growth at the points. Thus, the initial population always contains rules that are satisfied at least once. Every new rule is checked for already existing similar rules in the population. Fitness value is also calculated for each rule at the initialization step.

Selection for Recombination. Crossover, Mutation

Selection for recombination determines parental pairs for further crossover and mutation. In the research, there were implemented such methods of selection for recombination: panmixia, inbreeding, outbreeding and roulette-wheel selection. *Panmixia* – selection type, where any two random individuals have the same probiliity to form a parent couple. *Inbreeding* – selection type, where genetically similar chromosomes are preferred as parent couple. In *outbreeding*, individuals with genetically different encodings are preferred. *Tournament selection* is held by selecting random $1 \le d < N$ individuals from the population (N – population length) and choosing best individuals according to their fitness. *Roulette-wheel selection* is a stochastic selection, where probability of selection is proportional to the fitness function.

Implementation of inbreeding and outbreeding uses search of similar rules. Share of similar subrules is used as comparative characteristic of similarity.

The method of selection is the same throughout the run of the algorithm and is determined by model parameter.

Recombination consists of two stages: crossover and mutation. Crossover operators are implemented differently depending on data presentation, but, usually, offsprings inherit traits of both parents.

The algorithm uses one-point crossover, so crossing point v (subrule index) for rules Ω^1 and Ω^2 is selected randomly, and offsprings Ω^{12} and Ω^{21} are expressed as following:

$$\Omega^1 = if\left(w^1{}_1 \wedge w^1{}_2 \wedge \ldots \wedge w^1{}_v \wedge w^1{}_{v+1} \wedge \ldots \wedge w^1{}_{z-1} \wedge w^1{}_z\right) \quad then \quad w^1{}_c,$$

$$\Omega^2 = if\left(w^2{}_1 \wedge w^2{}_2 \wedge \ldots \wedge w^2{}_v \wedge w^2{}_{v+1} \wedge \ldots \wedge w^2{}_{z-1} \wedge w^2{}_z\right) \quad then \quad w^2{}_c,$$

$$\Omega^{12} = if\left(w^1{}_1 \wedge w^1{}_2 \wedge \ldots \wedge w^1{}_v \wedge w^2{}_{v+1} \wedge \ldots \wedge w^2{}_{z-1} \wedge w^2{}_z\right) \quad then \quad w^2{}_c,$$

$$\Omega^{21} = if\left(w^2{}_1 \wedge w^2{}_2 \wedge \ldots \wedge w^2{}_v \wedge w^1{}_{v+1} \wedge \ldots \wedge w^1{}_{z-1} \wedge w^1{}_z\right) \quad then \quad w^1{}_c.$$

Mutation procedure is held with probability p, defined as a model parameter. Indices s,k or rank b could be changed in a random subrule $\tilde{w} = \Delta(s,k) \in [y_b, y_{b+1})$ according to the allowed space of values.

Procedures of reconstructing rule indices with respect to the initial one, verification of similar subrules and fitness evaluation are performed after each recombination operation.

Survival selection. GA termination condition

After the recombination stage, offsprings are added to the population, if no similar chromosomes already exist, or they replace existing similar individuals, if their fitness value is higher. Note that in this implementation, there exists a threshold of minimal number of rule satisfactions in the time series.

Stopping criterion of the algorithm is defined as a sufficient number of run iterations I. GA produces a set of rules with a given estimation of quality – the value of the fitness function, which is used for forecasting.

Construction of the final forecast

Matching phase. Each rule is checked if it can be applied for each of the prediction position $\tilde{t}$. Rules are checked for those elements of the time series, where the conditional part belongs to known values of the time series, and "*then*" part belongs to the period of advance. If the conditional part of the rule is satisfied for a certain position t, then a possible forecast can be made using "*then*" part.

The result of matching phase is a set of conflicts $\{<\Omega_j, \phi, offset>\}$ of length $h_{\tilde{t}}$ for each prediction position $\tilde{t}$. The set contains information about the rule Ω_j, its fitness value ϕ and rule's *offset* on the time series. Thus, each position $\tilde{t}$ is corresponded to $h_{\tilde{t}}$ alternative values $\hat{G}$ of the forecast.

Conflict resolution phase. In the research, aggregated value across all conflicting values $\hat{G}$ is calculated considering priority of each rule (its fitness function). The above interpretation of the meaning of fitness contains information about the specificity of the rule and its satisfaction frequency in the time series.

Action phase. Input information of the forecasting algorithm contains sets of conflicts, time-series data and period of advance . The predicted value is calculated as:

$$\hat{F}_{\tilde{t}} = \frac{\sum_{i=1}^{i=h_{\tilde{t}}} \hat{G}_i \times \phi_i}{\sum_{i=1}^{i=h_{\tilde{t}}} \phi_i},$$

where $\tilde{t}$ – position in the advancing period for which the forecast is calculated, $\tilde{t} = \overline{1,P}$,

P – advancing period,

$\hat{F}_{\tilde{t}}$ – resulting value of the forecast for the position $\tilde{t}$,

$\hat{G}_i$ – conflicting value of the possible forecast for the position $\tilde{t}$,

$h_{\tilde{t}}$ – number of conflicts for the position $\tilde{t}$,

ϕ_i – fitness value of the rule that caused conflict $\hat{G}_i$,

$i = \overline{1,h_{\tilde{t}}}$.

Analysis of implementation issues of the forecasting algorithm

We developed specialized forecasting software package in language C#, using proposed approach. The calculations were performed on the virtual machine Microsoft Azure A8, designed for intensive computing, which has the following characteristics: processor Intel Xeon 2.5 Gh, 8 cores, 56 GB of RAM. To evaluate the efficiency of established software and algorithmic tools, we conducted computational experiment on the prediction of the real-world time series (Figure 3). We used monthly time series of actual sales of drugs in pharmacies in Ukraine for the period of 2004-2013 years, expressed in UAH. To achieve the goal of computing experiment, time series data were divided into a training sample (2004-2012 years), test sample (2012 year) and control sample (2013 year).

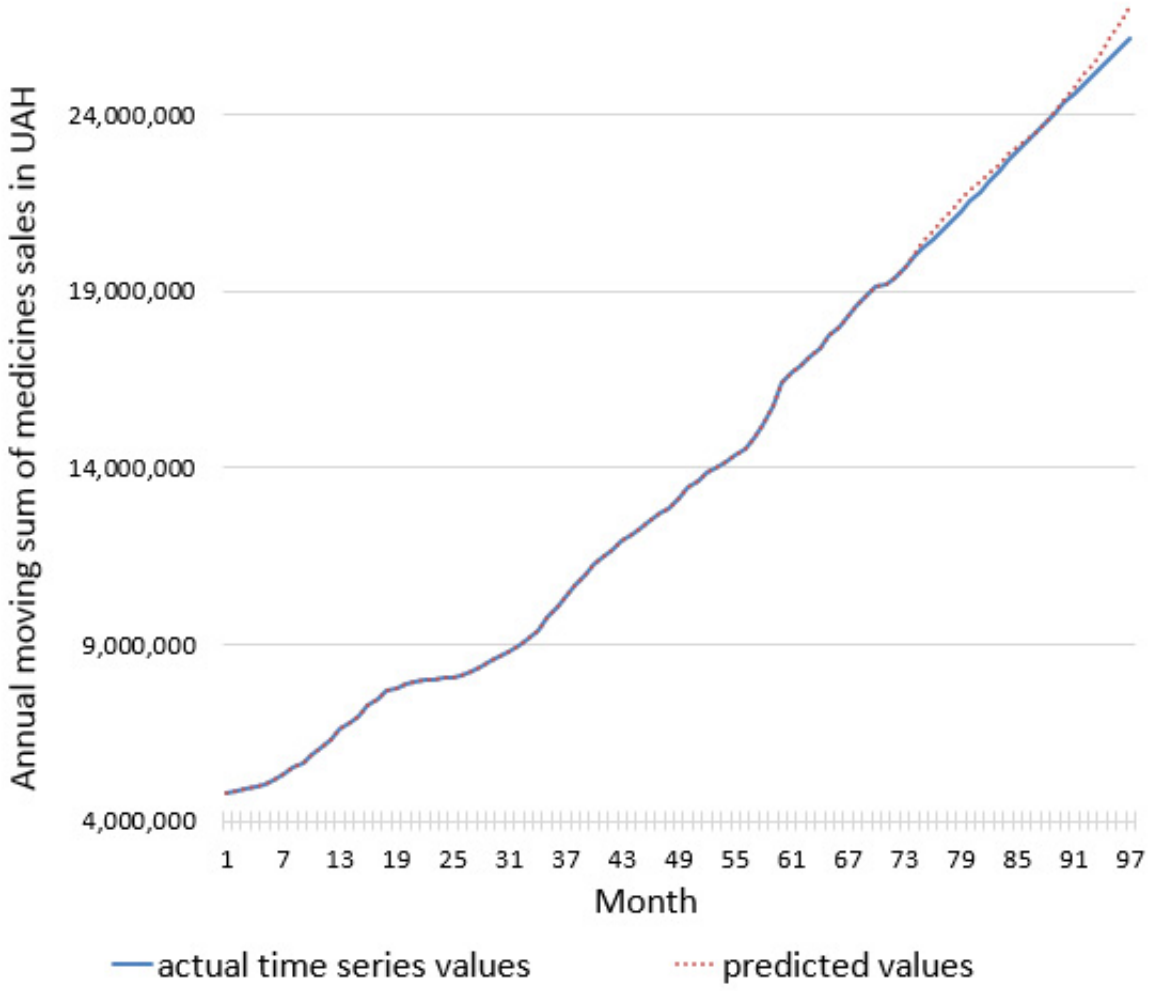


Figure 3. Forecast of annual moving sum of drug sales in UAH

Error evaluation is based on the known criteria MPE, MAPE [Hanke, 2008]:

$$MPE = \frac{1}{N}\sum_{t=1}^{N}\frac{\hat{F}_t - F_t}{F_t} \times 100\%,$$

$$MAPE = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{\hat{F}_t - F_t}{F_t}\right| \times 100\%$$

where $\hat{F}_t$ – predicted value,

F_t – actual value of the time series,

N – number of points for comparison (usually, the available time series),

t – time moment, $t = \overline{1,N}$.

Three sets were obtained from the time series: training set, test set and control set. The test set is used to validate correctness of the generated theory, that is, that the learning system has been able to model the concept, represented by instances in the training set, instead of modelling only instances themselves [Bacardit, 2004].

Table 1 shows the actual data for 2012th year, forecast for 2012th year (test sample), actual data for 2013th year and forecast for 2013th year (control sample), obtained via computational experiment.

Computational experiment shows that average values of forecast errors by the criterion of MPE on the test and the control samples were -1.1% and -1.2% respectively, while for the criterion MAPE – 1,1% and 1.2%, which is considered acceptable. [Гуляницкий Л.Ф., 2014b].

Table 1. Forecasting results

№	Actual values of the test sample	Forecast values of the test sample	Actual values of the control sample	Forecast values of the control sample
1	19949504	20059232	23228344	23295940
2	20206460	20387146	23494700	23487183
3	20463903	20712021	23765372	23762372
4	20721069	20981187	24051883	24069264
5	20993146	21297783	24323290	24412651
6	21261310	21593159	24567628	24754267
7	21530600	21851015	24828381	25108251
8	21817053	22098811	25091582	25418312
9	22105440	22312336	25350755	25794917
10	22393529	22604263	25609683	26224947
11	22679450	22882337	25876747	26606814
12	22961491	23044748	26141320	27063485
MAPE		**1.1%**		**1.2%**
MPE		**-1.1 %**		**-1.2%**

Forecasting algorithm's parameters settings and tuning

In the study, we carried on an experiment to determine optimal parameters of the algorithm. Number of conducted experiments for each combination was $K = 50$, the period of advance of the forecast was 12 months. Experiment used a smoothed time series – annual moving sum of medicines sales in UAH.

Figure 4 shows the dependence of the error MAPE, selection for recombination approach and minimal size z of the chromosome in the initialization phase with other conditions remaining the same ($\vartheta = 3$, $\varsigma = 3$, $\rho = 70\%$, $\gamma = 3$, $m = 100$, $\alpha = 0.8$, $\delta = 1$, $l = 100$, $\upsilon = 20\%$). It is obvious that with the growth of the initial size of the chromosome, error increases. This can be explained by the fact that during the growth of the chromosome size, number of rule satisfactions is reduced, and rules lose generalization feature. Additionally, it is necessary to analyze the possibility of simultaneous increase of the specifity of initial rules (size of chromosomes), the initial population size and the number of learning iterations.

Figure 5 represents the dependence of MAPE criterion, selection for recombination approach and number of iterations with other conditions remaining the same ($\vartheta = 3$, $\varsigma = 3$, $\rho = 70\%$, $\gamma = 3$, $m = 100$, $\alpha = 0.8$, $\delta = 1$, $z = 3$, $\upsilon = 20\%$). Figure 5 shows that the best MAPE value is reached at $l = 600$ for all selection methods. Rule set is not learnt enough within small number of iterations and becomes overlearnt (rule set converges quicklier and niching methods become less efficient) with growth of learning duration.

Figures 6-8 show dependence between initial rule length z (subrules count), initial population size and MAPE error.

It is obvious from figure 6 that aggregated MAPE error across all values of initial population size is the best for initial chromosome length z equal to four. Additional experiments showed that long rules have lower satisfaction frequency, therefore, their fitness is smaller. Figure 7 demonstrates that with growth of initial population size for fixed number of iterations MAPE value becomes higher. Figure 8 displays dependence between above parameters in three dimensions.

Figures 9-11 describe dependence between permissible subrule shifting error ϑ, permissible covered segment error ς and MAPE error. Both parameters ϑ and ς are used in niching procedures, but experiment results show that they are not correlated.

Figures 12-14 show dependence between permissible interval errors γ, total number of intervals $2m$ in time series and MAPE Error.

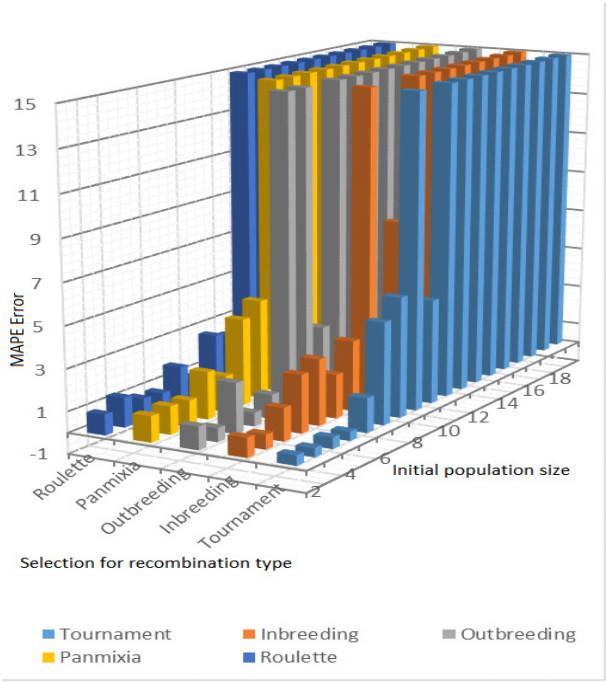


Figure 4. Dependence of the MAPE, selection for recombination approach and minimal size *z*

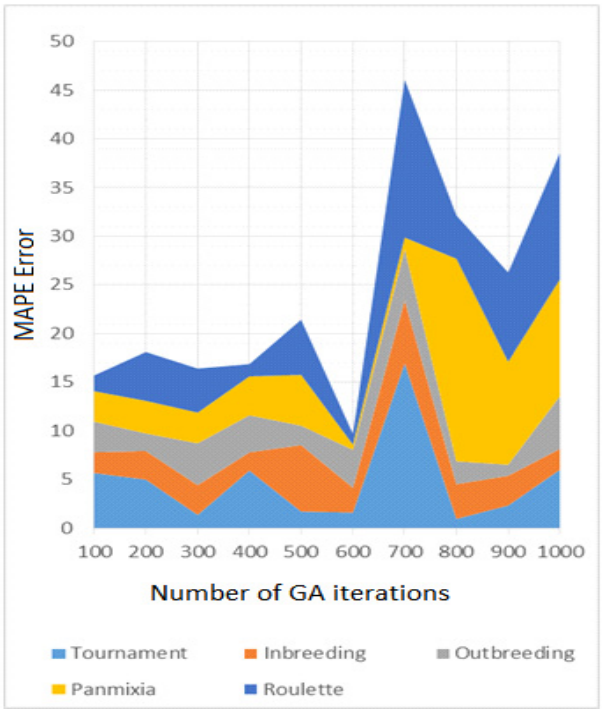


Figure 5. Dependence of the MAPE, selection for recombination approach and minimal size *z*

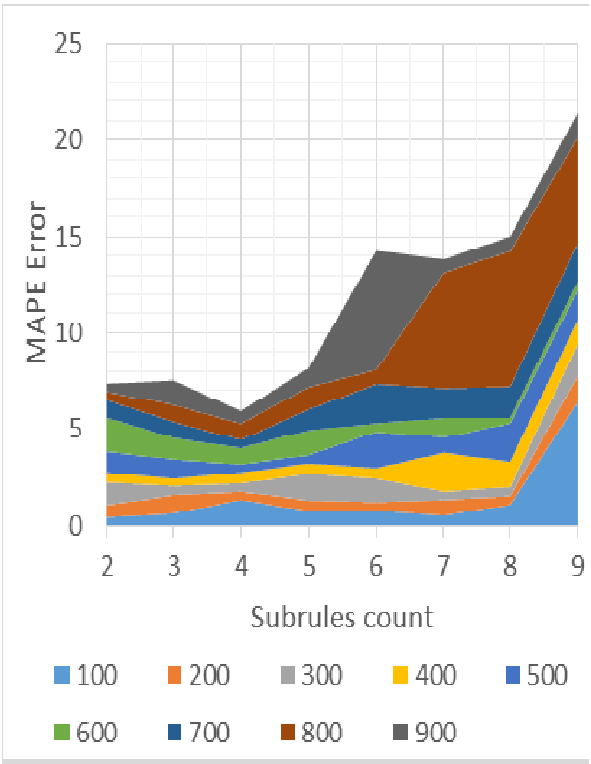


Figure 6. Dependence between subrules count, population size and MAPE

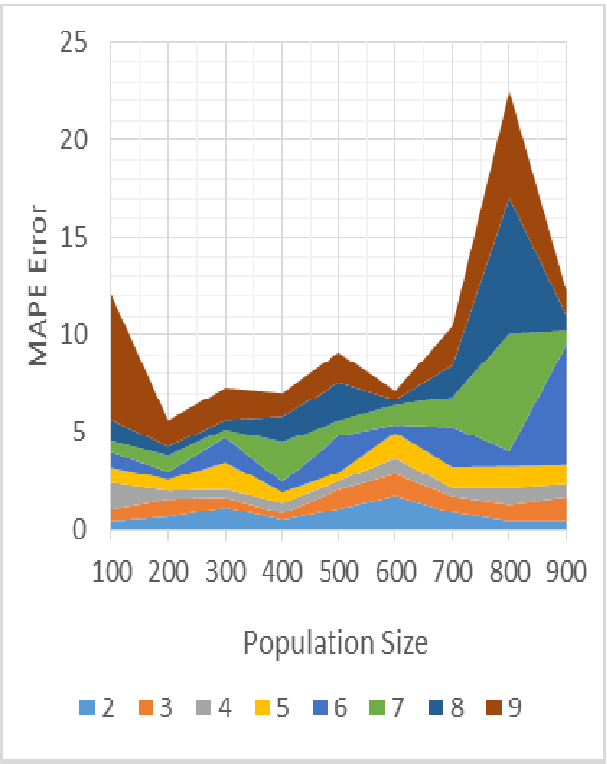


Figure 7. Dependence between subrules count, population size and MAPE

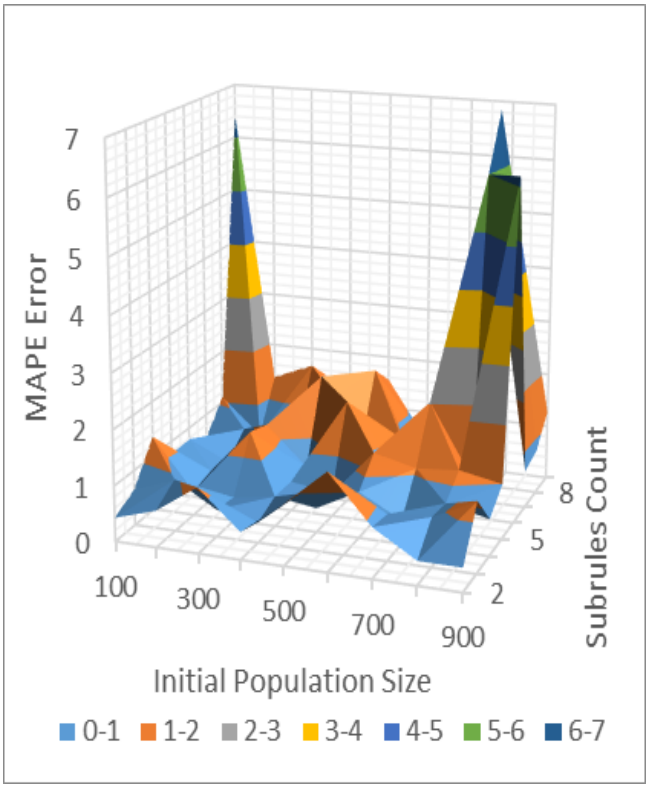


Figure 8. Dependence between subrules count, population size and MAPE

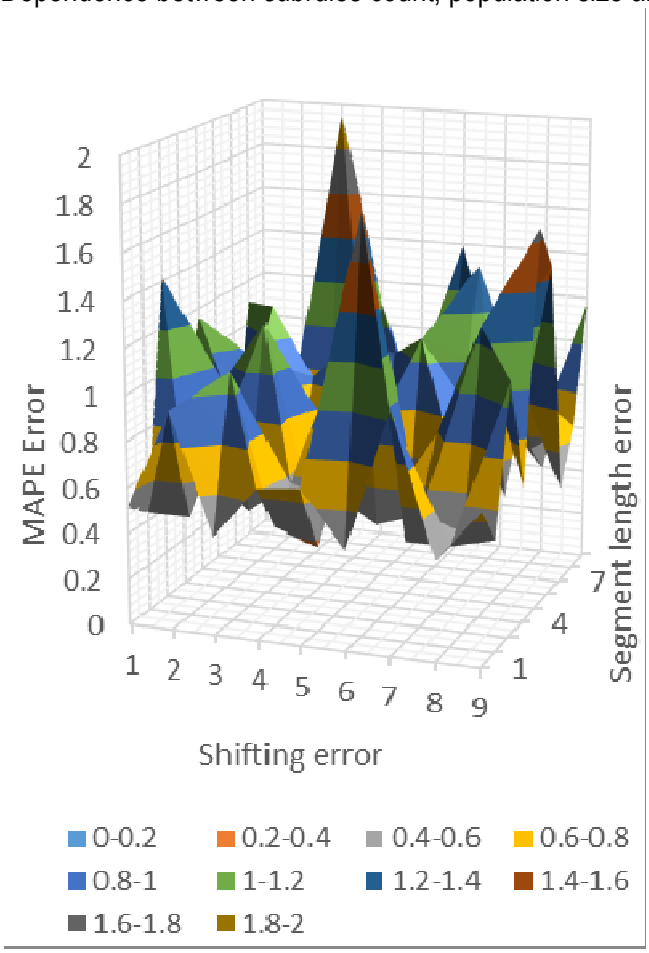


Figure 9. Dependence between permissible subrules' shifting errors, covered segment error and MAPE

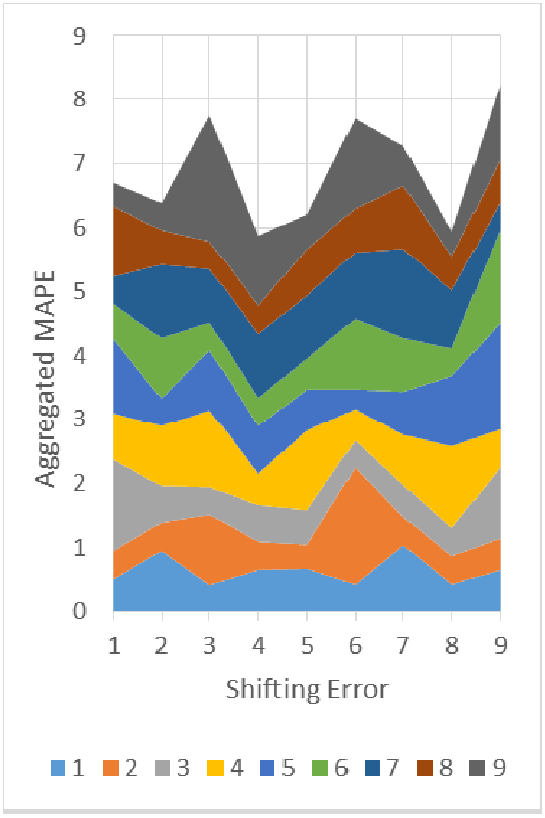


Figure 10. Dependence between permissible subrules' shifting errors, covered segment error and MAPE

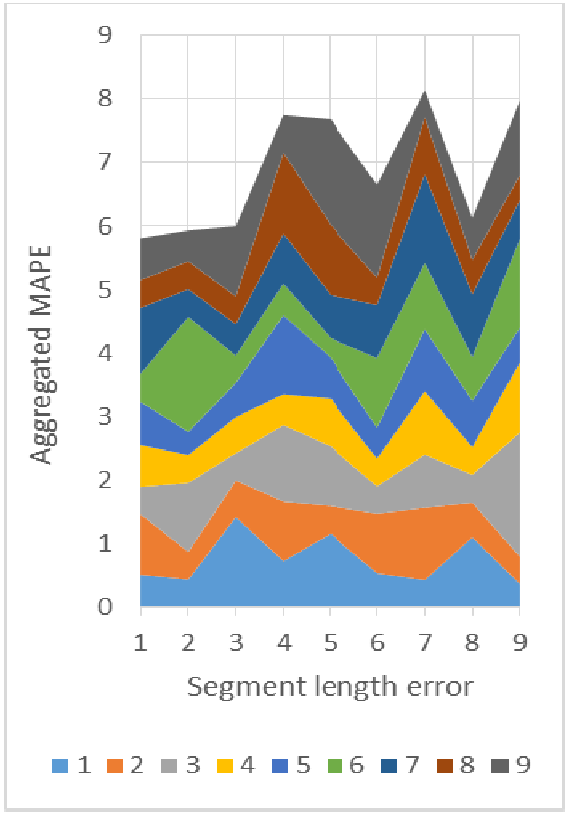


Figure 11.– Dependence between permissible subrules' shifting errors, covered segment error and MAPE

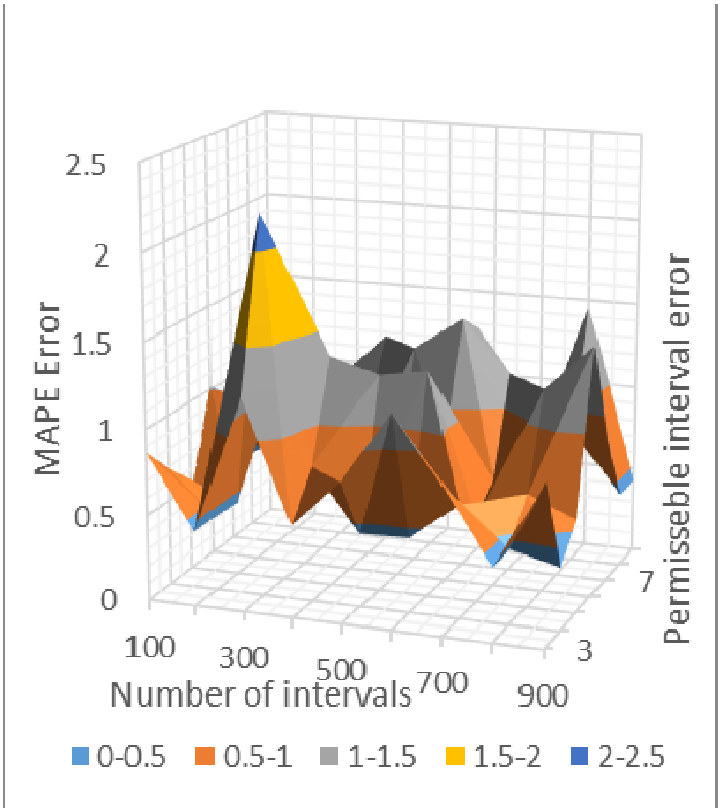


Figure 12. Dependence between permissible interval errors, total number of intervals in time series and MAPE

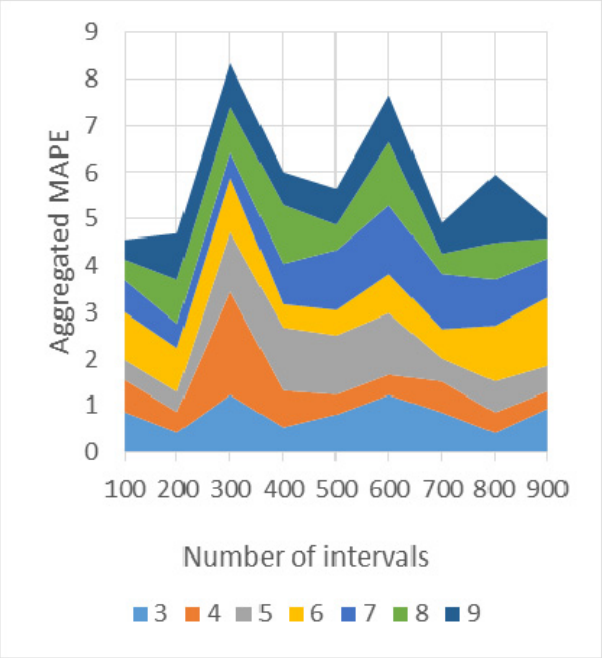


Figure 13. Dependence between permissible interval errors, total number of intervals in time series and MAPE

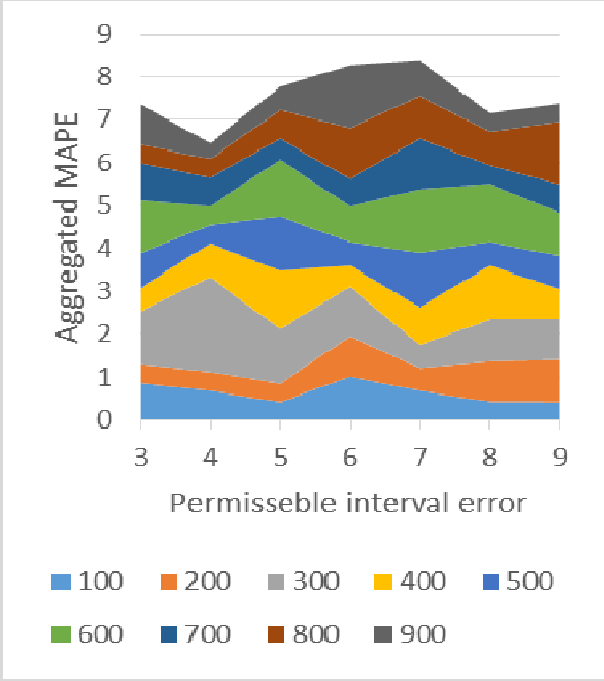


Figure 14/ Dependence between permissible interval errors, total number of intervals in time series and MAPE

Conclusion

This paper describes main stages of rule-based forecasting using GA. The proposed method of data representation allows applying the algorithm for forecasting problems with multiple time series. We also designed an approach to chromosome quality evaluation, which considers its length and satisfaction frequency. These measures of rules quality are used at the stage of conflict resolution and construction of the final forecast. A mechanism for the diversification of the search space was developed. Genetic operators for the proposed method of data presentation are described.

We conducted several computational experiments to select optimal values of algorithm parameters.

Further research requires to analyze issues of selection of optimal parameters of the algorithm and construction of the forecast with the influence of several factors.

Bibliography

[Kovacs, 2010] Tim Kovacs. Genetics-based Machine Learning. Handbook of Natural Computing: Theory, Experiments, and Applications. Springer Verlag, 2010. – 62p.

[Eiben A.E., 2007] Eiben A.E., Smith J. E. Introduction to Evolutionary Computing / Natural Computing Series .– Springer, 2007. – 300p.

[Bacardit, 2004] Jaume Bacardit. Pittsburgh Genetic-Based Machine Learning in the Data Mining era: Representations, PhD thesis, Ramon Llull University, Barcelona, Spain, 2004. – 352p.

[Sam Mahfoud, 1996] Sam Mahfoud, Ganesh Mani. Financial forecasting using genetic algorithms. – Applied Artificial Intelligence, 1996. – P. 543-560.

[Fernandez A., 2010] A Fernández, S García, J Luengo, E Bernadó-Mansilla, F Herrera. Evolutionary Computation, IEEE Transactions, 14 (6), 2010. – P. 913-941.

[Павленко А.І., 2014] Павленко А.І. Застосування генетичного алгоритму для прогнозування на основі бази правил / Компьютерная математика. - 2014. – P. 74-80.

[Sasikumar M., 2007] Sasikumar M., Ramani S., Muthu Raman S., Anjaneyulu KSR. A Practical Introduction to Rule Based Expert Systems. - Narosa Publishing House, New Delhi, 2007. – 294p.

[De Jong K.A., 1993] De Jong K.A., Spears W.M., Gordon D.F. Using genetic algorithms for concept learning.– Machine Learning, 1993. – P. 161-188.

[Гуляницкий Л.Ф., 2014a] Гуляницкий Л.Ф., Павленко А.И. Прогнозирование на основе генетического алгоритма обучения. - Конференция KDS, 2014. – P. 41-42.

[Гуляницький Л. Ф., 2014b] Гуляницький Л. Ф., Павленко А.І. Налаштування параметрів моделі алгоритму прогнозування на основі генетичних алгоритмів. Теорія прийняття рішень. – 2014. – P. 90-93.

[Mahfoud S.W., 1995] Mahfoud S.W. Niching methods for genetic algorithms. – Dissertation Abstracts International. – University Microfilms, 1995. – 251p.

[Goldberg D. E., 1987] Goldberg D. E., Richardson J. Genetic algorithms with sharing for multimodal function optimization. – In Genetic algorithms and their applications: Proceedings of the second international conference on genetic algorithms, 1987. – P. 41-49.

[Hanke, 2008] John E. Hanke, Dean Wichern. Business Forecasting. – Prentice Hall, 9 edition, 2008. – 576p.

Authors' Information

Hulianytskyi Leonid *– Dr.Sc. (Technology), Head of department of Glushkov Institute of Cybernetics of NAS of Ukraine, Professor of NTUU "KPI" (Kyiv); e-mail: leonhul.icyb@gmail.com*

Major fields of scientific research: combinatorial optimization; decision making; mathematical modeling and applications; forecasting.

Pavlenko Anna *– PhD student at Glushkov Institute of Cybernetics of NAS of Ukraine (Kyiv) Glushkov Ave., 40, Kyiv, 03680, Ukraine; e-mail: dmitrieva.anya@gmail.com*

Major fields of scientific research: forecasting, genetic algorithms, artificial intelligence; combinatorial optimization; mathematical economics.

SIMULATION MODELING IN THE CONSTRUCTION OF DYNAMIC INTEGRATED EXPERT SYSTEMS

Galina Rybina, Victor Rybin

Abstract: *Problems of integrating methods and simulation modeling tools with the technology of dynamic integrated expert systems (IES) developed on the basis of task-oriented methodology and AT-TECHNOLOGY workbench are investigated in the context of the modern stage of simulation modeling development. The current version of the simulation modeling subsystem, functioning as a part of the dynamic version of the AT-TECHNOLOGY workbench, is described. Basic subsystem components and the construction technology of simulation models of complex technical systems are considered, as well as examples of application of this subsystem in prototyping dynamic IES are provided.*

Keywords: *dynamic intelligent systems, dynamic integrated expert systems, task-oriented methodology, simulation, complex engineering systems, simulation modeling subsystem, simulation model, AT-TECHNOLOGY workbench.*

Introduction

Analysis of modern works in the scope of simulation modeling conducted in [1] shows a significant increase of interest in this area, which, experiencing a rebirth, is becoming more and more advanced computer simulation technology used in a wide range of new applications related to the control and decision-making of the technological, organizational, economic, social and other character in dynamic problem domains [2-5].

Today the integration technology with data mining (Data Mining) [6, 7], computer-aided design [8, 9], business information systems [10, 11] etc. are actively investigated in simulation modeling problems in addition to the traditional aspects (methodological, mathematical, technological and applied).

However, integration processes achieved the greatest topicality in dynamic intelligent systems (DIS) [12, 13], such as intelligent control systems [14, 15], intelligent decision support systems [16, 17], dynamic integrated expert systems [6, 12], multi-agent systems and intelligent agents [18-21], that connected with an constantly-expanded range of DIS application in modern postindustrial society - from the spacecraft and sophisticated automated manufacturing and robots to organizational and technical (socio-technical) systems with uncertain human factor.

It should be noted that the problems of integration of simulation modeling technology with dynamic integrated expert systems (IES) got the most complete development under task-oriented methodology and supporting software tools (AT-TECHNOLOGY workbench) [6]. Scientific and practical results in the theory and technology for constructing IES published in the monograph [6], as well as repeatedly described in various scientific journals [22-27], including the pages of some issues of this journal [28-30].

The focus of this article which is a continuation of the researches described in [6, 22-33] is technological and applied aspects connected with the expansion of the architecture of the dynamic version of the AT-TECHNOLOGY workbench by integrating the subsystem simulation modeling of the external world and the combined functioning of this subsystem with temporal solver and other basic workbench components in the development of dynamic IES prototypes.

Features of the Application of Simulation Modeling in the Construction of Dynamic IESS

As has been noted in a number of papers, for example [6, 22], long-term experience of practical use of task-oriented methodology and AT-TECHNOLOGY system has shown, that the applications in the which problem area is dynamic occur very often, and therefore a cycle of researches has begun and led to the creation of theory and software technology constructing dynamic IESs which operates in real time, i.e. IESs using dynamic representation of object domains and solving dynamic problems (main results were published in [1, 28-36]).

In the context of the further development of task-oriented methodology for other architectures of DIS, in particular, multi-agent systems also conducted researches related to the creation of simulation systems interaction of intelligent agents (IMVIA), which is currently used as a workbench developing the prototypes of multi-agent systems for dynamic problem fields (described in [12, 20] and other papers).

Most often, the most common areas of the application of dynamic IESs technology proved complex engineering systems (CES) and complex engineering and organizational systems (CEOS), for which the classical simulation focused on formal mathematical models was practically impossible to use. In [6] introduced a specific definition, under which CES and CEOS - the following objects of a technical nature: their parameters constantly vary (in real time); they comprise from several hundred to several thousand functionally and structurally interrelated components, subsystems, modules, units, etc.; the diagnostics of these objects can be considered as a specific control process with the goal of determining the technological state of objects at each current instant (the general task of diagnostics of the object status) and, in addition, the task of fault finding (as a special case of the general diagnostic task); the functioning of these objects is a complex technological process accompanied by a multitude of abnormal conditions, rapid changes in the environment, and the lack of time for decision-making in response to abnormal conditions; a high price is paid for errors made by operators.

Accordingly the dynamic IESs for discrete and continuous-discrete CES/CEOS must ensure, in the general case, support for the execution of the following tasks: the dynamic modeling of all processes of functioning the of CES/CEOS; monitoring the CES/CEOS operation, detection of deviations from the prescribed regime, pre-failure alerting and abnormal condition warning, emergency cut-out, etc.; studying the actions of the operators who control CES/CEOS and training of personnel; a convenient graphic user interface for monitoring variations in the basic parameters characterizing CES/CEOS operation, etc.

In the context of the use of the task-oriented methodology for constructing IESs [6], the additional functionality of described tasks entails a significant change of the IES architecture as all basic components of static IES are practically modified, especially, knowledge base and reasoning tools, and two new subsystems are added—subsystem modeling the external world (environment) and subsystem interfacing with the physical equipment, as well as the technology of constructing dynamic IESs is significantly changed. The subsystem interfacing with the external environment is necessary to obtain a constant data stream from external equipment and sensors, and the subsystem modeling the external world (environment) is intended to simulate the data stream at all stages of the life cycle of dynamic IES development.

In the context of this work, the subject of discussion is the subsystem modeling the external world, because the data that is transferred to working memory by the subsystem uses temporal and universal solvers of the AT- TECHNOLOGY workbench [6, 27, 31] to realize a deduction and to obtain recommendations. Basing on these objectives for computer simulation of the CES / CEOS behavior in time the simulation modeling concept using RAO-approach [37], which implements the process-oriented approach to construct simulation models (SM) achieved the most development and application. As the expansion experience of the AT-TECHNOLOGY workbench architecture by specialized tools in the form of the simulation modeling subsystem of the external world which is realized on the basis principles of the RAO-approach [31-36] has shown, this way was quite effective for deep integration of all components of the dynamic IES nucleus and combined functioning of the simulation modeling subsystem with the temporal solver, universal AT-SOLVER and other basic components of the dynamic version of AT- TECHNOLOGY workbench (individual results discussed in [1, 27-36]).

The basic principles of the simulation modeling subsystem implementation based on RAO-approach and task-oriented methodology requirements are considered in more detail.

General Characteristic of the Simulation Modeling Subsystem

- ✓ *Main Principles of the Construction of the Simulation Modeling Subsystem*

In the architecture of the simulation modeling subsystem, the functionality of the developed tools is divided between two global modules [6, 31] – the “SM development module” whose tasks are to support

the development process and debugging of SM and other functions requiring the visual interface, and the “SM computation module” ensuring the computation of the conditions of SM in each time step (cycle) of the functioning process of the dynamic IES.

The development of a powerful full-featured high-level language to describe the SM and the creation of a corresponding compiler for that language are the unifying conceptual framework for the two basic modules. To implement this approach, at the first stage of the researches, formalism RAO is used as a language to describe the SM, the basic version of which is given in [37]. In the future, based on the analysis of current requirements to design models of CES / CEOS to create dynamic IESs developed a special language “RAOAT” including new conceptual changes associated with object -oriented language and significant technological expansions due to the addition of new instructions and data storage structures [31–36].

Let us briefly consider RAOAT language features. The basic concept of the creation of RAOAT language is its full object orientation, and therefore, such basic principles of object-oriented language, as encapsulation, inheritance and polymorphism are implemented. From the standpoint of encapsulation should be noted that into the language were entered modifiers for resource types that allow to describe the internal logic of resource types that change the state of the resource attributes according to some conditions (logical or temporary). Also entered the internal attributes of the resource types that are not available for changing by operation templates, that it significantly reduces the effort required in the design and development of SM containing a set of resource types that are represented by sets of an arbitrary number of instances by transferring the description of variation of resource conditions from the operation templates to the state modifiers of resource types.

Inheritance and polymorphism in the RAOAT language are represented with the ability to inherit one type of a resource from another and override the implementation of the state modifiers of a resource in the process of inheritance that can largely simplify the SM containing the sets of structurally similar resources.

It should be pointed out, that basic principles and implementation features of the several versions of simulation modeling subsystem in the context of the proposed approach have been repeatedly described in different papers, in particular [31–36]. In general, it should be noted that such basic requirements, as the ability to create new objects and resource types on the basis of finished, the ability of developing model base through the development of resource types library, the efficiency in developing and maintaining complex SM (hundreds or thousands of objects and resource types, well-developed logic), the suitable RAOAT language allowing to construct SM by specialists on simulation modeling rather than programmers and the efficient ways to integrate with third party software, were considered when developing the subsystem.

Another important feature is related to the “SM development module” ensuring the user (knowledge engineer and / or specialist on simulation modeling) by convenient visual editor that allows *operating* graphical objects and relationships between them within the concept of RAO-method. Meanwhile, a code generation of the model in RAOAT language occurs (established resource types, resource templates operations, etc.). Result of the visual design of the model is the description of the model in RAOAT language. The tools of this module allow you to construct a model of the outside world for dynamic IES using visual, flexible, extensible, repeated-used objects (resource types). The ability to increase the life cycle of SM by rapid adjustment to changing conditions, when solving of which require both high and low levels of abstraction, is achieved due to an unlimited nesting of resource types and inheritance .

Composition and structure of the “SM development module” and the “SM computation module” functioning in the structure of the current version of simulation modeling subsystem (dynamic version of the AT-TECHNOLOGY workbench) [1, 31-36] are briefly considered bellow.

- ✓ *Architecture, Composition and Structure of the Basic Components of the Simulation Modeling Subsystem*

General architecture, composition and structure are of the basic components of the current version of simulation subsystem, detailed description of which is given below.

Visual Objects Editor: The component “Visual objects editor” allows you to create objects, setting their properties and attributes, as well as establish relationships between the model objects in graphical mode. Knowledge engineer and / or specialist on simulation modeling, operating with the models editor, designs SM on graphic canvas containing a predetermined number of various objects, at that the ability to create, delete, copy objects and set relationships between them is ensured. The values of a particular set of properties are able to change for each object. The created visual representation of SM and properties of all objects are stored in the memory of the editor, as well as in a separate text file, which is further processed by the RAOAT language compiler. This visual tool allows you to load a saved SM to update the visual presentation and insert the changes manually to obtained code which is described in RAOAT language.

Models Synthesis Component: The models synthesis component, interacting with the visual objects editor by processing the stored collections of objects, generates a description of SM in RAOAT language in XML format which is passed to the RAOAT language compiler.

Component of Visualizing SM: Knowledge engineer, if necessary, with the support of the “Animation frames and displayed rules editor” selects the description of the model in RAOAT language and makes animation frames for corresponding objects, and the tool “Visualizer" based on the values of resource

parameters, descriptions of animation frames and displayed rules ensures drawing the animation frames.

RAOAT Language Compiler: The obtained description of SM in RAOAT language is passed to the input of the "SM computation module" where the compilation from RAOAT language to C# language occurs and the further interpretation and run the developed model. The kernel of the "SM computation module" is the "RAOAT language compiler" which structure has a standard form for the syntax-driven three-pass compiler. This compiler consists of an analyzer which includes components of the lexical, syntactic and semantic analysis, and synthesis component including a generation component of output code. Features of the implementation of several versions of the RAOAT language compiler are described in detail in [31, 32, 34]. It should be noted that the availability of such objects in the RAOAT language, as irregular events and temporary resources, requires the time coordination with each object of the model. Each object has its own internal timer showing within the modeling time scale, how much time obtains an object. In addition, this timer is associated with the total time of an activity of the SM for a corresponding pause during transferring and obtaining the data from the working memory of the temporal solver.

Supporting Component of Computing the SM States: The "Supporting component of computing the SM states" ensures the generation of a discrete modeling time, as well as the generation of the control actions used to start or stop the activity of SM in the form of messages on each discrete time step. Computation of the new state of SM is based on the state at the previous time step and an allowance of executed operations.

Technology of Constructing Simulation Models Using Simulation Modeling Subsystem Tools

A technology for constructing SM described as five suitable interacting stages is considered in detail.

Stage 1. Description of CES/CEOS. First, the knowledge engineer and/or specialist on simulation modeling (expert) selects the simulation objects (resources) and resource parameters, and then the rules for interactions between resources (events) are determined and the detachment of events to regular and irregular is executed with the indication of the relevant event resources.

Stage 2. Constructing the description of SM in RAOAT language. A knowledge engineer and / or a specialist on simulation modeling constructs the description of RAOAT objects (resource types, resources, templates of operations, operations, functions), and a program control of the correctness of the description of RAOAT objects (object parameter blank, infringement of the logic of object, etc.) is realized. Each described object is added to the corresponding collections, and then models synthesis component generates a conversion of the object collections to the description of SM in the RAOAT format (saving in corresponding XML-structure is realized).

Stage 3. Compilation of SM in RAOAT language to the internal representation of C# language. With the help of RAOAT compiler generates a compilation of SM in RAOAT language of XML- structure. The description of SM in RAOAT language passes lexical analysis stage, result of which is the formation of object lists, then the stage of parsing, at which the resulting lists of objects are converted to the parse tree, and at the stage of semantic analysis realized the checking of correctness of the description of SM in accordance with the logic of RAOAT language. Further, the synthesis component of target text realizes a transformation of original description of SM to the code in C# language, and the result description is compiled into .dll.

Stage 4. Parameters configuration and computing of SM states

4.1. Configuration of SM parameters. The obtained SM in the form of .dll enters the input of the "Supporting component of computing SM states" to be realized the configuration of SM parameters using specialist on simulation modeling (time step length, number of time steps). After completing the configuration of SM parameters, SM is executed, and then control of the SM can be realized by stopping or pausing it.

4.2. Computation of SM states. Computation of the states of SM is executed based on the state of SM at the previous time step, with an allowance for the operations executed by SM. If the time step is not the last, then computing the states of SM to the next time step is realized on the basis of resource parameter values in the current tact. The activity of the component comes to complete after obtaining the last time step and the file which contains the values of resource parameters at all time-step is generated. In case of if the visual examination of the state of SM is not required, and result of the activity of the SM is suited by the specialists, then the first iteration is passed, i.e. the first version of SM is obtained. If the visual examination of SM is needed, then transition to the next stage 5 is realized.

Stage 5. Visualization of SM. The "Component of visualizing SM", which consists of two units – the "Animation frames and displayed rules editor" and the "Visualizer", is activated when the visual examination of the state of SM is necessary. Editor obtains a description of SM in RAOAT language as an input, and the knowledge engineer and / or a specialist on simulation modeling makes adding animation frames (the path to the image, width, height, etc.) and displayed rules that determine the behavior of the animation frames. Added animation frames and displayed rules are written in the corresponding XML- structure and passed to the input of "Visualizer", together with the values of resource parameters at all time-step for the graphical representation of SM. If the obtained graphical representation on the results of the activity of SM is suited by the specialists, then the first iteration is passed, i.e., the first version of SM is obtained.

Technological cycle is completed at this and the achieved version of SM in the form of .dll which is intended for further use in the process of prototyping of dynamic IESs is transferred to the working memory by special tools of AT- TECHNOLOGY workbench which obtain the SM in the form .dll ensure the synchronization of the processes of interacting between simulation modeling subsystem and reasoning tools (temporal solver and AT-solver). As the temporal solver, as well as the SM, operates on time step, the interaction is realized through the exchange of data and commands in asynchronous mode [35]. At the end of each time step of the activity of SM, data transferring to the solver through the working memory is realized. The obtained data initializes a new reasoning cycle. From now the solver has a certain time quantum allowing realizing a deduction for urgent situations. The resulting urgent commands (the deduction of which had time to execute within the time limit) are used at the next time step of SM. Not considered urgent and common situations are processed before the end of the next time step and used through a time step of the activity of SM. In case of obtaining new data from SM a deduction is processed and delivered an interim solution. The next time step of the activity of temporal solver does not depend on the place where the deduction of previous time step has broken down.

Conclusion

Thus, the further development and deep integration of the methods of simulation modeling with the methods of constructing dynamic IESs in the context of task- oriented methodology as a unified framework ensuring the complex decision for scientific and applied problems related to the construction of different classes of dynamic IESs, allowed to create an approach that does not have analogues today. This approach has been successfully implemented in the framework of the dynamic version of AT- TECHNOLOGY system and has already shown its efficiency in implementation of several prototypes of applied dynamic IESs.

Acknowledgements

This work was supported in part by the Russian Foundation for Basic Research under Grant No. 15-01-04696

Bibliography

1. Рыбина Г.В., Рыбин В.М., Со Ти Ха Аунг. Некоторые аспекты применения имитационного моделирования в динамических интегрированных экспертных системах // Приборы и системы. Управление, контроль, диагностика. 2014. № 3. С. 35-45.
2. Плотников А.М., Рыжиков Ю.И., Соколов Б.В. Современное состояние и тенденции развития имитационного моделирования в Российской Федерации // Пятая всероссийская научно-практическая

конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика». Сборник докладов. Т.1. – СПб.: ОАО «Центр технологии и судостроения», 2011. С.55-61.

3. Лычкина Н. Н. Инновационные парадигмы имитационного моделирования и их применение в сфере управленческого консалтинга, логистики и стратегического менеджмента // Современные технологии управления логистической инфраструктурой - III: Сборник научных статей, – М.: Изд-во Эс-Си-Эм Консалтинг, 2012. С. 9-28.
4. Averill M., Kelton W. Simulation modeling and analysis. 3rd ed. McGraw-Hill,Inc., 2000. – 760 p.
5. Banks J. Handbook of Simulation: Principles, Methodology, Advances, Applicationsand Practice. – Wiley, 1998. – 864 p.
6. Рыбина Г.В. Теория и технология построения интегрированных экспертных систем. Монография. – М.: Научтехлитиздат, 2008. − 482с.
7. Rybina G.V., Deineko A.O., Distributed Knowledge Acquisition for the Automatic Construction of Integrated Expert Systems // Scientific and Technical Information Processing. 2011.V. 38. № 6. P.1-7.
8. Таратухин В.В. Интеграция систем имитационного моделирования и экспертных систем в САПР // КИИ-98. Шестая национальная конф. с международным участием. Сб. научных трудов в 3-х томах. – Пущино: РАИИ, 1998. Т.2. С. 461-465.
9. Seidel S., Donath U., Haufe J. Towards an integrated simulation and virtual commissioning environment for controls of material handling systems //Proceedings of the 2012 Winter Simulation Conference. 2012. P. 2852 -2863.
10. Лычкина Н.Н. Современные технологии имитационного моделирования и их применения в информационных бизнес-системах и системах поддержки принятия решений // Вторая всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика». Сборник докладов. Т.1. – СПб.: ФГУП ЦНИИ технологии судостроения, 2005. С. 26-31.
11. Pidd M. Computer simulation in Management Science. 5th ed. Chichester:Wiley, 2004. – 328 p.
12. Рыбина Г.В., Паронджанов С.С. Технология построения динамических интеллектуальных систем: Учебное пособие. М.: НИЯУ МИФИ, 2011. − 240с.
13. Осипов Г.С. Методы искусственного интеллекта. – М.: ФИЗМАТЛИТ, 2011. – 296с.
14. Макаров И.М., Лохин В.М., Манько С.В., Романов М.П. Искусственный интеллект и интеллектуальные системы управления. – М.: Наука, 2006. − 333с.
15. Рыбин В.М. Интеллектуальное управление на основе динамических интегрированных экспертных систем. // Информационно-измерительные и управляющие системы. 2011. №6. С. 16-19.
16. Вагин В.Н., Еремеев А.П. Некоторые базовые принципы построения интеллектуальных систем поддержки принятия решений реального времени // Известия РАН. ТиСУ. 2001. №6. С. 114-123.

17. Башлыков А.А., Еремеев А.П. Семиотические системы реального времени для интеллектуальной поддержки принятия решений при управлениями сложными технологическими объектами // Информационно-измерительные и управляющие системы. 2013. №5. С.49-57.
18. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. – М.: Эдиториал УРСС, 2002. − 352с.
19. Fougères A. Modelling and simulation of complex systems: an approach based on multi-level agents // IJCSI International Journal of Computer Science Issues, November 2011.Vol. 8, Issue 6, No 1, P. 8-17.
20. Rybina G.V., Parondzhanov S.S. Modeling of Intelligent Agent Interactions for Multiagent Systems // Scientific and Technical Information Processing. 2010. V. 37. № 5. P. 318-327.
21. Аксенов К.А., Смолий Е.Ф. Мультиагентное имитационное моделирование бизнес-процессов и организационно-технических систем, среда BPSIM // Пятая всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика». Сборник докладов. Т.1. – СПб.: ОАО «Центр технологии и судостроения», 2011. С.11-19
22. Рыбина Г.В. Практическое применение задачно-ориентированной методологии построения интегрированных экспертных систем (обзор приложений в статических и динамических проблемных областях) // Приборы и системы. Управление, контроль, диагностика. 2011. № 12. С. 10-28.
23. Рыбина Г.В. Интегрированные экспертные системы // Приборы и системы. Управление, контроль, диагностика. 2011. №4. С. 22-47.
24. Рыбина Г.В. Технология построения интегрированных экспертных систем на основе задачно-ориентированной методологии // Приборы и системы. Управление, контроль, диагностика. 2011. №6. С. 28-42.
25. Рыбина Г.В. Теоретические основы задачно-ориентированной методологии построения интегрированных экспертных систем. // Приборы и системы. Управление, контроль, диагностика. 2011. №7. С. 27-42.
26. Рыбина Г.В. Модели и методы реализации интеллектуальной технологии построения интегрированных экспертных систем. // Приборы и системы. Управление, контроль, диагностика. 2011. №10. С. 27-37.
27. Рыбина Г.В. Инструментальный комплекс АТ-ТЕХНОЛОГИЯ для поддержки построения интегрированных экспертных систем: общая характеристика и перспективы развития. // Приборы и системы. Управление, контроль, диагностика. 2011. №11. С.17-40.
28. Рыбина Г.В., Демидов Д.В., Шанцер Д.И., Мозгачев А.В. Динамические интегрированные экспертные системы: новые возможности инструментального комплекса АТ-ТЕХНОЛОГИЯ // Информационно-измерительные и управляющие системы. 2011. №6. С. 7-15.
29. Рыбина Г.В., Мозгачев А.В., Со Ти Ха Аунг. Построение динамических интегрированных экспертных систем на основе задачно-ориентированной методологии // Информационно-измерительные и управляющие системы. 2012. №8. Т.10. С. 4-12.

30. Рыбина Г.В. Экспертные системы и инструментальные средства для их разработки: некоторые итоги // Информационно-измерительные и управляющие системы. 2013. Т.11. № 5. С. 35-48.
31. Рыбина Г.В. Инструментальные средства для построения динамических интегрированных экспертных систем: развитие комплекса АТ-ТЕХНОЛОГИЯ // Искусственный интеллект и принятие решений. 2010. №1. С. 41-48.
32. Rybina G., Shantzer D.I., Mozgachev A.V. Some aspects of evolution of tools for support dynamic integrated expert systems development // Proceedings of First Russia and Pacific Conference on Computer Technology and Applications (RPC 2010). 6-9 September 2010, Vladivostok (Russia). Vladivostok: IACP FEB RAS, 2010. P. 136-139.
33. Рыбина Г.В., Шанцер Д.И., Мозгачев А.В., Блохин Ю.М. Динамические интеллектуальные системы на основе интегрированных экспертных систем // Приборы и системы. Управление, контроль, диагностика. 2012. №5. С. 13-19.
34. Рыбина Г.В., Шанцер Д.И., Липатов А.А. Имитационное моделирование внешнего мира при построении динамических интегрированных экспертных систем// Тринадцатая национальная конференция по искусственному интеллекту с международным участием. Труды конференции. Т.3. – Белгород: Изд-во БГТУ, 2012.С.179-185
35. Рыбина Г.В., Мозгачев А.В., Пароджанов С.С., Со Ти Ха Аунг Динамические интегрированные экспертные системы: методы представления и обработки темпоральных знаний // Приборы и системы. Управление, контроль, диагностика, 2013. №6. С.63-73
36. Rybina G.V., Rybin V.M., Parondzhanov S.S., SoeThi Ha Aung. Some aspects of simulation application in dynamic integrated expert systems // Life Science Journal. 2014. V.11. №8s. P.145-149.
37. Емельянов В.В., Ясиновский С.И. Введение в интеллектуальное имитационное моделирование сложных дискретных систем и процессов. Язык РДО. − М.: АНВИК, 1998. – 533с.

Authors' Information

Galina Rybina *– Doctor of Technical Science, Professor cybernetics department of NRNU MEPhI. RF President education award winner. Accents on intelligent systems and technologies, static, dynamic and integrated expert systems, intelligent dialogue systems, multi-agents systems, workbenches; e-mail: galina@ailab.mephi.ru*

Victor Rybin *- Doctor of Technical Science, Professor department of Automation of NRNU MEPhI. Accents on automation and electronics, electro physical complex, automatic control system, intelligent control systems, dynamic intelligent systems; e-mail: vmrybin@yandex.ru*

ONTOLOGICAL APPROACH TO A CONSTRUCTION OF THE SIMULATION SYSTEM FOR THE SPECIFIC DOMAIN

Elena Zamyatina, Alexander Mikov, Roman Mikheev

Abstract: *This paper discusses the problem of simulation system constructing for the specific domain. Authors suggest using the ontological approach. The simulation system TriadNS is considered. This simulation system is dedicated for computer networks design and analysis. Authors represent base ontology and some other ontologies describing the concepts of a specific domain.*

Keywords: *simulation, ontology, computer network, multi model approach*

ACM Classification Keywords: *I.6 Simulation and Modeling, I.6.8 Types of Simulation - Distributed: I.2 Artificial Intelligence, I.2.5 Programming Languages and Software - Expert system tools and techniques*

Introduction

Simulation system is wide spread and well known method used to analyze the complicated dynamic systems (to design and to analyze the computer networks, or computing systems, or logistic systems, for example). Very often this method may be the only one for the investigations (if it does not have any analytical decisions). But the use of simulation may be more effective if the specialists in particular domains take part in these investigations. Indeed it is very rational to use graph theory to study computer networks (path finding). The traffic exploration may be carried out with the help of queueing theory; in order to explore the parallel and distributed systems it often requires to know the theory of Petri Nets. So it is expedient to make the usual (standard) terms and concepts available for the investigator in a specific domain. Ontology is a very convenient tool here because it consists of a set of concepts and a set of a links between these concepts.

The paper considers one more approach. It is based on the language workbench. Moreover, if it is necessary to investigate a problem, situation or other complicate systems more precisely one has to create different models and to transform these models [Sokolov & Yusupov, 2005]. So an investigator must work with the set of models. The set of models allows improving the quality of the investigations and makes simulation experiment more efficient (qualimetry of simulation model).

Ontologies in simulation and simulation systems

Ontology is the description of types of entities of a specific domain, their attributes and relations between the entities. One can describe every specific domain (as a part of real world) by ontologies. Ontologies are created and used in various specific domains, and it is well known how to use the ontologies in the simulation.

But the creation of ontologies for simulation is a rather complicated problem and it is not easy to solve it because the simulation may be applied to various specific domains (chemistry, physics, logistic, transport and etc.). Moreover simulation is based on the mathematics, statistics, and the theory of probability (and so on), so their ontologies have to be the basic ones.

Ontologies may be used on the various stages of simulation beginning from the stage of data collection onwards (data about a specific domain) and accomplishing with the stage of validation [Sargent, 2005]. One can consider some ontology driven program tools dedicated to simulation (High Level Architecture, for example) and the problem of the alliance of the federates (in HLA) [Rathnam & Paredis, 2004]. This approach uses ontologies to describe the demands to which the federator's interface must correspond; what is needed for their successful cooperation. Moreover the ontologies are used to construct these demands with due regard to particular domain.

The paper [Liang & Pardis, 2003] presents the ontology of ports. This ontology can be considered as a mean for automatic building of a simulation model containing several components. Ports describe the interfaces of the components of a model or the interfaces of the subsystems in the system configuration. Overall system may be considered as the several subsystems or components connected to each other via well-defined interfaces. Ontologies are used successfully in other cases and this fact is reflected in various papers [Benjamen et al, 2005; Benjamen et al, 2006].

Setting to the specific domain

So it is necessary to set a simulation system to the specific domain. In fact it is the description of a simulation model with the help of specific concepts and relations between the concepts. And if an investigator uses graphical editor it is possible to describe a simulation model with the help of one of visual languages.

It is possible to use special program tools of language workbenches (or DSM-platforms), dedicated to creation of DSL (Domain Specific Language), to the setting simulation system to the specific domain. The second approach – program tools, based on ontologies. Let us consider the first approach.

Language workbench and DSL creation

There are various language workbenches (MetaEdit+, DSL Tools and so on) allowing to create meta models (DSL). These meta models permit to create models of a specific domain. Meta-language [Сухов, 2013] is one of the language workbenches and it was designed and implemented in Perm State University. Designers of meta-language removed the shortcomings of similar program tools. The design of simulation system set to the specific domain consists of several steps:

- It is creation of a new model, every model has a name and sometimes it is necessary to describe the model's attributes (if required). Meta-language has a graphical editor for model building. It is necessary to define the basic construction of the language. The basic elements in Metalanguage are entities, relations, restrictions. So an investigator defines basic entities of a meta model (DSL), relations between these entities and imposes a restrictions of the meta models;
- Afterwards an investigator creates models containing particular entities of specific domain and relations between these entities with the help of DSL;
- It is necessary check the validity of a model, the investigator has to be sure that the model satisfies all restrictions. If some of these restrictions are violated the investigator may be informed about this fact.

Program tools mentioned above may be applied not only for setting a simulation model to a specific domain and be applied to particular simulation system. These program tools have a wider application; the main application is the business model creation [Zamyatina et al, 2013].

Setting to specific domain using ontological approach

The second approach of simulation model setting to a specific domain is model-oriented approach, based on the ontologies. This approach is used, for example, to build a variety of GUI software [Gribova & Cherkezshvili, 2010].

The setting of simulation system to a specific domain may be done with the help of the ontologies for the following reasons:

- TriadNS has basic ontology and ontologies which are needed for the setting to a specific domain. Moreover ontologies may be needed for the completeness of partly defined simulation model. The complete model may be built due to including the program fragments (routines) into simulation model. Special program component TriadBuilder searches these fragments with the help of ontology in data base of routines using semantic types;
- It is well known that computer networks are an example of a rapidly changing specific domain (new types of networks, new protocols, new algorithms of message exchange and new routing algorithms appear). It is therefore necessary to have special linguistic and software tools

allowing to describe these new devices and algorithms and so to perform an automatic or semiautomatic setting to a specific dynamically changing domain;

- Computer networks can include a large number of computing nodes, so the simulation requires a large amount of time. The way out is the parallelization of a simulation run, distribution of objects of the simulation model by the nodes in intensive computing environment. At that, it is necessary to keep the causality of events;
- It is important to optimize simulation system in the respect of time because computer networks may consist of tremendous number of nodes. So it is necessary to use the resources of several computing nodes of mainframe, or cluster (and etc.) in order to reduce the overall time of simulation experiment. Simulation model in this case must be presented as a set of logical processes in this case. These processes are distributed between computing nodes usually. Processes communicate one another during simulation run, classical synchronization algorithms (optimistic and conservative) govern the functioning of logical processes. Efforts of many authors focus on improving these algorithms. In TriadNS also developed improved synchronization algorithms (based on the optimistic one). Feature of these algorithms is that they use the knowledge of the model, and this knowledge is partially extracted from the ontology. Moreover, it is necessary to have equal workloads on the computing nodes during simulation run. Special component of simulator named Triadbalance provides load balancing during simulation experiment. It is designed as multi-agent systems. Agents act according to the rules extracted from ontologies both basic and associated with particular subject area [Mikov et al, 2013];
- Simulation model verification and validation may be carried out using ontologies too [Zamyatina et al, 2013]. Ontologies contain the rules and restrictions. Simulation model has to be built in accordance with these rules and restrictions;
- It is convenient to integrate components automatically into simulation model or to include the new components from the other simulation models and systems into a simulation model.

So we can conclude that ontologies may be used at every stage of simulation of computer network. So it is rational to use them for setting to a specific domain. Investigations show that ontologies allow making simulation system adaptable and providing it is openness [Zamyatina & Mikov, 2012].

Let us consider the principles of simulation model construction in TriadNS, linguistic and software tools for collecting data during simulation run and etc. before the precisely consideration of ontologies in TriadNS.

Simulation model representation in TradNS

Computer simulator was designed and implemented on the base of CAD TRIAD [Mikov, 1995].

Simulation model in TriadNS is represented by several objects functioning according to some scenario and interacting with one another by sending messages. So simulation model is μ={STR, ROUT, MES}

and it consists of three layers, where STR is a layer of structures, ROUT – a layer of routines and MES – a layer of messages appropriately. The layer of structure is dedicated to describe objects and their interconnections, but the layer of routines presents their behavior. Each object can send a message to another object. So, each object has the input and output poles (P_{in} – input poles are used to send the messages, P_{out} – output poles serve to receive the messages). One level of the structure is presented by graph P = {U, V, W}. P-graph is named as graph with poles. A set of nodes V presents a set of programming objects, W – a set of connections between them, U – a set of external poles. The internal poles are used for information exchange within the same structure level; in contrast, the set of external poles serves to send messages to the objects situated on higher or underlying levels of description. Special statement <message> through <name of pole> is used to send the messages.

One can describe the structure of a system to be simulated using such a linguistic construction:

structure <name of structure> ***def*** (<a list of generic parameters>) (<a list of input and output parameters>) <a list of variables description> <statements>) ***endstr***

Special algorithm (named “routine”) defines the behavior of an object. It is associated with particular node of graph P = {U, V, W}. Each routine is specified by a set of events (E-set), the linearly ordered set of time moments (T-set), and a set of states {Q-set}. State is specified by the local variable values. Local variables are defined in routine. The state is changed if an event occurs only. One event schedules another event. Routine (as an object) has input and output poles (Pr_{in} and Pr_{out}). An input pole serves to receive messages, output – to send them. One can pick out input event e_{in}. All the input poles are processed by an input event, an output poles – by the other (usual) event. The special statement ***out*** (***out*** <сообщение> ***through*** <имя полюса>) serves for sending a message.

A set of routines defines a routine’s layer ROUT.

The layer of messages (*MES*) is necessary for a description of a message with complicate structure.

Simulation algorithm is a set of the objects which function in accordance to their scenarios, objects send messages to one another and they are governed with the help of special program – “simulator”. Each routine has local calendar of the scheduled events. Each scheduled event has a time stamp (a scheduled moment of time when event will occur). Simulator has to find the smaller time stamp (among all local calendars) and call the appropriate routine containing the event with the smaller time stamp.

But it is very important to assess the simulation model and it is behavior in defined conditions and in accordance to appropriate restrictions.

Data Collection and Processing of Data During simulation run

Special program component – the *condition of simulation* defines the scenario of simulation experiment, the criterions of simulation run termination, a list of simulation model elements (the variables, the events, the input and output messages) which are have to be examined and processed during simulation run with the help of the *information procedures* and the scenarios of the completing processing of the results of these information procedures. A component "c*onditions of simulation"* contains a set of another program of "information procedures" needing for data collection and processing of data during simulation run.

Conditions of simulation and information of procedures present an "algorithm of investigation".

Simulation system Triad includes a library of standard information procedures but an investigator may describe an information procedure (and conditions of simulation too) with the help of Triad language:

information procedure<*name*>*(*<*a list of generic parameters*>*)(*<*input and output formal parameters*>*)*

initial <*a sequence of statements*> ***endi*** <*a sequence of statements*> ***processing*** <*a sequence of statements*>*…****endinf*** {it is the description of information procedure}.

The linguistic construction of **c*onditions of simulation:***

Conditions of simulation <*name*>*(*<*a list of generic parameters*>)(<input and output formal parameters>) **initial** <a sequence of statements> ***endi*** <a list of information procedures> <a sequence of statements> ***processing*** <a sequence of statements>…***endcond***

Simulation experiment

The investigation of the structure layer only is static process. The simulation process may take place only after the definition of the behavior of all nodes of model in structure layer. As it was noted above the behavior is determined by the statement ***Put***. It is well known that a simulation is a set of object functioning according to some definite scenarios controlled by synchronizing algorithm. The simulation run is initialized by the statement ***simulate***:

Simulate <a list of models> ***on condition of simulation*** <имя условия моделирования>(<настроечные параметры>)(<параметры интерфейса>)(<список информационных процедур>; <последовательность операторов>).

One can pay an attention to the fact that the several models may be simulated under the same conditions of simulation simultaneously.

Moreover it is possible to define several parts of linguistic construction "*conditions of simulation"* in the statement *simulate*. It is rather important. Let us suppose that an investigator wish to design computer network, so it is necessary to define the structure of computer network and characteristics of workstations and other devices of computer network. Let us suppose that an investigator want to define the configuration of computer network and technical characteristics of cheapest network, of more efficient network, of more secure one and etc. So it is necessary to define different criteria during simulation run and one can do it using different "conditions of simulation" in statement simulate.

Ontological approach to multi model simulation

It is important to involve into the simulation process not only the specialists in simulation but the specialist in specific domains and specialists in the other spheres of knowledge. That is why it is necessary to adjust a simulation system to specific domain. Indeed the investigator of computer network may use a graph theory while studying the structure of network, or a queue network theory, or the theory of Petri Nets. Ontologies are used in TriadNS to set the simulation system to specific domain. Let us briefly consider the decision of this problem in TriadNS.

Basic ontology

Ontologies can be applied on the different stages of simulation. Very often ontologies are applied for the simulation model assembly [Liang & Pardis, 2003]. So the simulation model may consist of separately designed and reusable components. These components may be kept in repositories or may be found via Internet. The ontologies keep the information about interconnections of simulation model components and other characteristics of these components.

It is well known that ontologies enable investigators to use one and the same terminology, so it is one more argument for the use of ontologies. The basic ontology is designed in TriadNS[Mikov et al, 2007].

Its basic classes are: TriadEntity (any named logic entity), Model (simulation model), ModelElement (a part of simulation model and all the specific characteristics of a node of a structure layer), Routine (node behavior), Message (note, please, that nodes of simulation model can send message to any other message) and so on.

The basic properties of the basic ontology in TriadNS are:

– *The property of ownership:* model has a structure, a structure has a node, a node has a pole and so on*;*

- *The property to belong to something* –– inverse properties to previous one– The structure belongs to the model, the node belongs to structure, the pole belongs to the node and so on;
- *The properties of a pole and an arc connection:* connectsWithArc (Pole, Arc), connectsWithPole (Arc, Pole);
- *The property of a node and an appropriate routine binding*-putsOn (Routine, Node);
- *The properties of a node and an appropriate structure binding:* explicatesNode (Structure, Node), explicatedByStructure (Node, Structure);
- *The property of the model and conditions of simulation binding* (Model, ModelingCondition).

The fragment of basic ontology is present below (see Figure1):

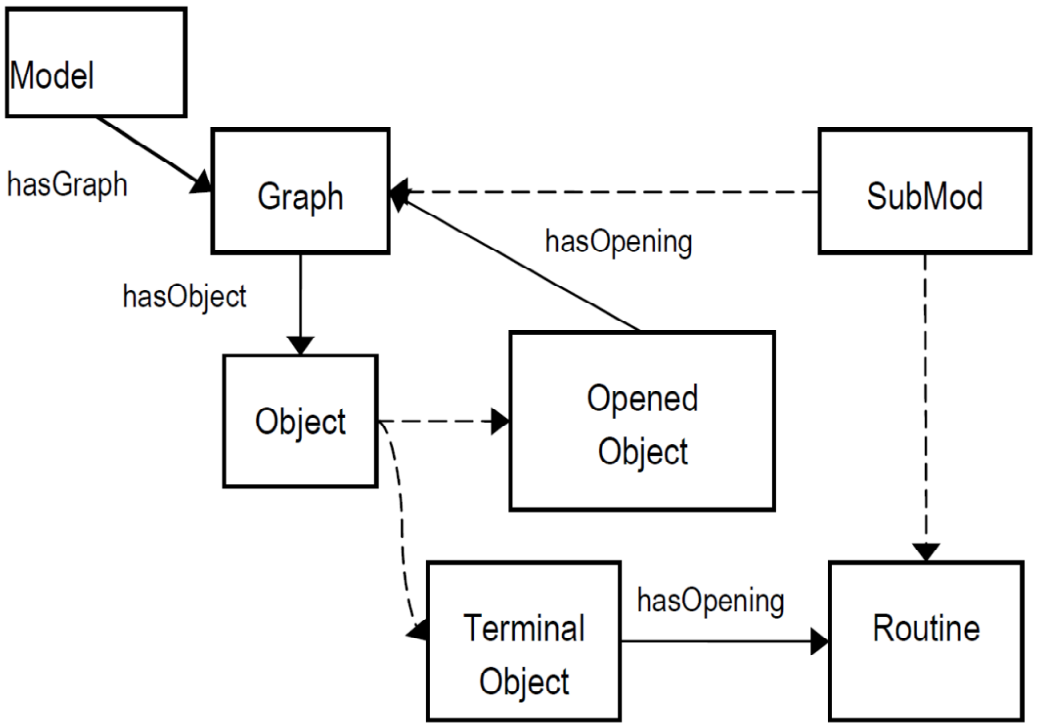


Figure 1. Basic ontology in TriadNS

The basic ontology is presented by the language OWL in simulation system TriadNS. OWL was chosen because there is a big number of programming tools for OWL processing, for the publication of ontologies in the Internet, for searching information (maybe appropriate ontologies) via Internet. Authors use Jena OWL API.

Program tools TriadNS uses ontologies in several cases: (a) for the completeness of (as it mentioned above, simulation model has a structure layer (a several of nodes and edges connecting these nodes) and a routine layer (every node has a behavior defining by routine. One may name simulation model as

incomplete or partly defined if any node haven't pointer to a routine defining it. Let us name such a node as "incomplete node"); (b) for integrating simulation models; (c) for construction simulation model in accordance to a specific domain (the main subject of this paper).

Let us describe the process of the completeness of partly defined simulation model in TriadNS more precisely:

- Special subsystem TriadBuilder [Mikov & Zamyatina, 2010] attempts to search the appropriate routine by the help of base ontology (it was described earlier). It may be found thanks to special semantic type (semantic type "Router" and "Host", for example);
- Model completion subsystem starts when the internal form of simulation model is not built according to a Triad code (it has incomplete nodes);
- First, module "model analyzer" searches all incomplete nodes of a simulation model, and marks them. After the *inference* module starts looking for an appropriate routine instance for each of marked nodes according to specification condition (the semantic type of node and routine must coincide).Then the condition of configuration must be checked (the number of input and output poles of node and the number of poles of routine must coincide). Third condition – the poles of incomplete nodes must be connected with the poles of appropriate nodes;
- So the simulation model can be completed by appropriate routines.

Ontology for a simulation of computer networks

The simulator TriadNS has some additional special subclasses of the base classes (specific domain – computer networks) [Zamyatina et al, 2011]:

- *ComputerNetworkModel* (a model of a computer network*), ComputerNetworkStructure* (a structure of a computer network model)*;*
- *ComputerNetworkNode* (a computer network element, it contains several subclasses: Workstation, Server, Router);
- *ComputerNetworkRoutine* (a routine of a computer network) and so on.

This ontology includes two special properties of a pole. These properties are used to *check* the conditions of matching routine to a node:

- *isRequired(ComputerNetworkRoutinePole, Boolean)* – this property check if it is necessary to connect a pole with another pole?
- *canConnectedWith(ComputerNetworkRoutinePole, ComputerNetworkRoutine)* – this property check the semantic type of an element of a structure being connected.

Ontology representing the models of computer networks as the queuing network

The ontology of semantic types of the queuing network supplements the basic ontology in the following way. Firstly, subclasses of the basic ontology classes specific for a queuing network domain are included: MpsModel – a model of the queuing network, MpsStructure – structure of the model of the queuing network and so on.

Secondly, four subclasses of the node class which correspond to the basic elements of the queuing network are included: Generator, Queue, Channel, Terminator.

Third, the classes representing the information specific for the queuing network are included: class of arrival time distribution, class of a service time distribution and so on.

Hierarchy of semantic types in ontology is represented as a hierarchy of routine classes and its top element is subclass of Routine class of the basic ontology - MpsRoutine that is a routine of the queuing network.

When constructing the hierarchy of semantic types the main attention is paid to the multiple inheritances.

The Figure 2 shows a fragment of the queuing network.

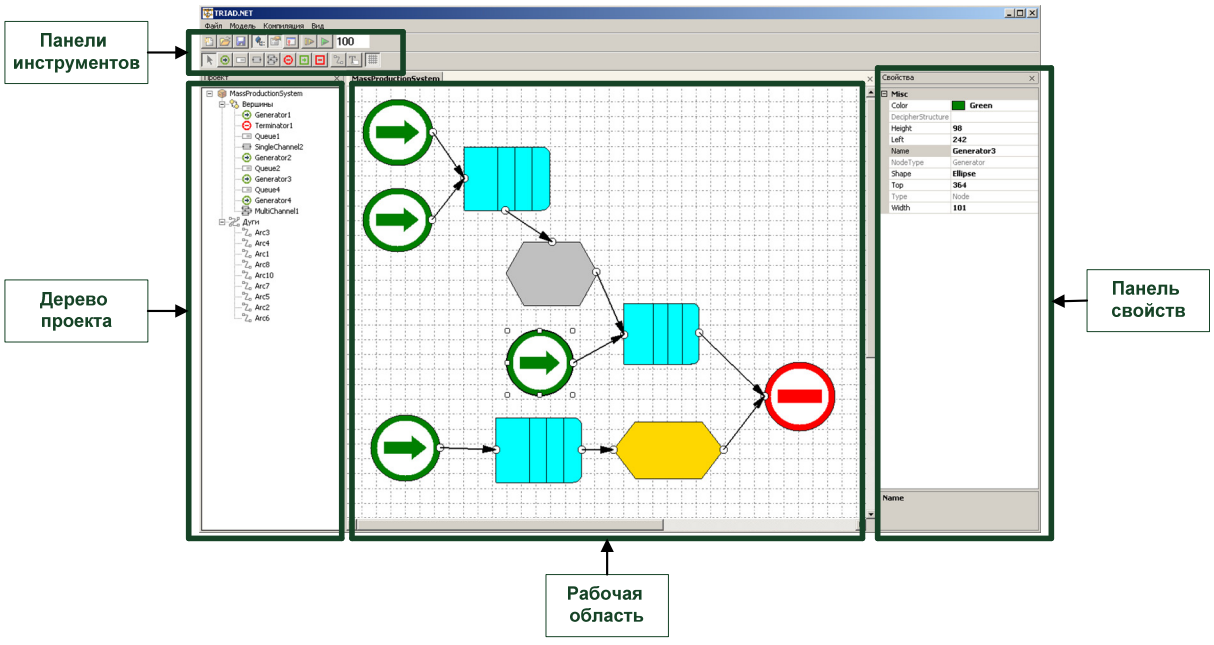


Figure 2. A fragment of queueing network

Conclusion

The paper demonstrates the ontological approach how to adjust the simulation system to a specific domain in the computer network simulator TriadNS.

An adjustment to a specific domain allows improving the quality of the simulation researches because the specialists from the different fields of knowledge can be invited, at that, they can work within their own domains using their standard notions, concepts and terms.

At present the basic and subject-oriented simulation systems are built for computer networks and queueing networks in TriadNS.

The authors of the paper are planning to design program and linguistic tools to transform one model into another (multimodel simulation).

Acknowledgments

The reported study was supported by RFBR, research project No. 13-07-96506_r-s-a “Design of deontic foundation of multiagent technologies of knowledge base construction for virtual laboratories”

Bibliography

[Benjamin et al, 2005] Benjamin P., Akella K.V., Malek K., Fernandes R. “An Ontology-Driven Framework for Process-Oriented Applications”// Proceedings of the 2005 Winter Simulation Conference / M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds., pp. 2355-2363.

[Benjamin et al, 2006] Benjamin P., Patki M., Mayer R. J. “Using Ontologies For Simulation Modeling”// Proceedings of the 2006 Winter Simulation Conference/ L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds., pp.1161-1167

[Gribova & Cherkezshvili, 2010] Грибова В.В., Черкезишвили Н.Н. “Развитие онтологического подхода для автоматизации разработки пользовательских интерфейсов с динамическими данными // Информационные технологии.-2010.-No10.-С. 54-58.

[Liang & Pardis, 2003] Liang V.C, Pardis C.J.J. “A Port Ontology for Automated Model Composition”// Proceedings of the 2003 Winter Simulation Conference / S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds., pp. 613-622.

[Mikov & Zamyatina, 2010] Миков А.И., Замятина Е.Б. “Проблемы повышения эффективности и гибкости систем имитационного моделирования”, Проблемы информатики, №4(8), Новосибирск, Институт вычислительной математики и математической геофизики СО РАН, 2010, С. 49-64.

[Mikov, 1995] Mikov A.I. “Formal Method for Design of Dynamic Objects and Its Implementation in CAD Systems” // Gero J.S. and F.Sudweeks F.(eds), Advances in Formal Design Methods for CAD, Preprints of the IFIP WG 5.2 Workshop on Formal Design Methods for Computer-Aided Design, Mexico, Mexico, 1995, pp.105-127.

[Mikov et al, 2007] Mikov A., Zamyatina E., Kubrak E. “Implementation of simulation process under incomplete knowledge using domain ontology” //Proceedings of the 6-th EUROSIM Congress on Modeling and Simulation, University of Ljubljana, Slovenia, 2007, Vol.2, Book of full papers, p.1-7.

[Mikov et al, 2013] Mikov A., Zamyatina E., Kozlov A., Ermakov S. “Some Problems of the Simulation Model Efficiency and Flexibility”, Proceedings of “2013 8th EUROSIM Congress on Modelling and Simulation EUROSIM 2013”, Cardiff, Wales, United Kingdom, pp. 532-538

[Rathnam & Paredis, 2004] Rathnam T., Paredis C.J.J. “Developung Federation Object Models Using Ontologies”// Proceedings of the 2004 Winter Simulation Conference / R .G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, eds., pp. 1054-1062.

[Sargent, 2005] Sargent R.G. “Some Recent Advances In The Process World View”// Proceedings of the 2005 Winter Simulation Conference / M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds., pp. 293-298.

[Sokolov & Yusupov, 2005] Соколов Б.В., Юсупов Р.М. “Концептуальные основы квалиметрии моделей и полимодельных комплексов” // Имитационное моделирование, Теория и практика: сб. докл. 2-й всерос. науч.-практ. конф. ИММОД–2005, Т. 1. СПб: ЦНИИТС, 2005, С. 65–70.

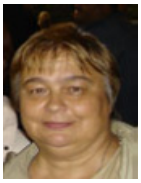
[Сухов, 2013] Сухов А.О. “Инструментальные средства создания визуальных предметно-ориентированных языков моделирования” / Фундаментальные исследования, 2013, № 4 (ч. 4), С. 848-852.

[Zamyatina et al, 2011] Замятина Е.Б., Миков А.И., Михеев Р.А. “Применение онтологий в системе проектирования и моделирования компьютерных сетей TRIADNS”, Труды Конгресса по интеллектуальным системам и информационным технологиям, IS&IT’11, Научное издание в 4-х.томах, М. Физматлит, 2011, Т.1, стр. 253-260

[Zamyatina & Mikov, 2012] Замятина Е.Б., Миков А.И. “Программные средства системы имитации Triad.Net для обеспечения ее адаптируемости и открытости”, Информатизация и связь, 2012, № 5, С. 130–133.

[Zamyatina et al, 2013] Замятина Е.Б., Лядова Л.Н., Сухов А.О. О подходе к интеграции систем моделирования и информационных систем на основе DSM-платформы MetaLanguage//В кн.: Технологии разработки информационных систем ТРИС-2013. Материалы IV Международной научно-технической конференции/ Т.1.Таганрог: Издательство Технологического института ЮФУ, 2013.с.61-70

Authors' Information



Elena Zamyatina *– Assoc. Prof. PhD, National Research University Higher School of Economics, Department of Business Informatics; Russia, Perm, 614070, Studencheskaya st., 38;*
e-mail: e_zamyatina@mail.ru
Fields of Scientific Research: simulation, artificial intelligence, ontology, domain specific languages (DSL)

Alexander Mikov *- Full Professor, Kuban State University, 149, Stavropolskaya str., Krasnodar, Russia, Computing Technologies Department, e-mail: alexander_mikov@mail.ru*
Fields of Scientific Research: information systems, simulation, artificial intelligence, ontology

Roman Mikheev *- “PROGNOZ”, Stakhanovskaya St, 54, Perm, Russan Federation, Programmer, e-mail: miheev@prognoz.ru.*
Fields of Scientific Research: Simulation, Artificial Intelligence, Computer Networks Simulation, Ontologies

ADAPTIVE ALGORITHM FOR MANAGEMENT BY WEIGHT COEFFICIENTS OF THE TRAFFIC IN CROSSBAR COMMUTATOR

Kiril Kolchakov, Vladimir Monov

Abstract: *An adaptive algorithm for management by weight coefficients of the traffic in Crossbar commutator is synthesised. The algorithm is adaptive because the matrix of weights corresponds to the matrix of the requests, each request receives a weight. The algorithm is modified in order to reduce the number of the weights by the diagonal location.*

Keywords*: Network nodes, Crossbar switch, Conflict elimination, Packet messages.*

ACM Classification Keywords*: C.2.1 Network Architecture and Design, C.4 Performance of Systems.*

Introduction

The purpose of this study is to find an approach for the conflicts problem solution in switching nodes of type Crossbar by management of weight coefficients of the traffic. For this purpose, an adaptive algorithm is synthesized for management by weight coefficients of the traffic in Crossbar commutator. The algorithm is adaptive because the matrix of the weights corresponds to the requests matrix. To each request a weight coefficient is assigned. The high weight corresponds to a high priority in the implementation of service requests.

The traffic via Crossbar switching nodes is casual and depends on the users. The formulation of a conflict issue during operation of the switching nodes is as follows. The dimensions of switches in the switching nodes are N x N, where N sources of packet massages are connected to N receivers via the switch of the switching node. The traffic is random by nature and conflicts are available in the following two cases:

1. When one source of message requests communication to two or more message receivers.
2. When one message receiver receives communication requests from two or more message sources.

The evasion of conflicts is directly related to the switching node performance. The status of the switch of the switching node is represented with the so called connection matrix. For N x N dimensional switch the dimension of the connection matrix T is N x N also, where every element $T_{ij} = 1$ if the connection request from i- source to j- receiver exists. In the opposite case $T_{ij} = 0$.

A conflict situation arises if any row of the connection matrix has more than a single 1, which corresponds to the case when one source requests a connection with more than one receiver. The presence of more than a single 1 in any column of the matrix T also indicates a conflict situation it means that two or more sources have requested a connection with the same receiver [Kolchakov and Monov, 2014a].

Approach for Traffic Control Description

The essence of the approach is to give the weight factors of the keys of the switch in order to avoid conflict situations to some extent.

The high weight corresponds to the high priority. It is applied a conflict indicator of requests (C), which is the ratio of the number of conflicts (K) in the matrix of connections (T) and the total number of requests (R).

$$C = K/R.100[\%] \quad (1)$$

Conflict can be caused by two or more service requests. The total number of requests is occasional by nature and depends on the traffic at the moment. Weight factors of the keys in the switch are changed periodically in a random law. The number of weights may vary. The matrix of the weights W is N x N size, like the connections matrix T.

Figure 1 shows an exemplary placement of the weight coefficients in W matrix with dimension 4 x 4 and five of the weight factors. Matrix T has 4 x 4 dimension also [Kolchakov and Monov, 2014a].

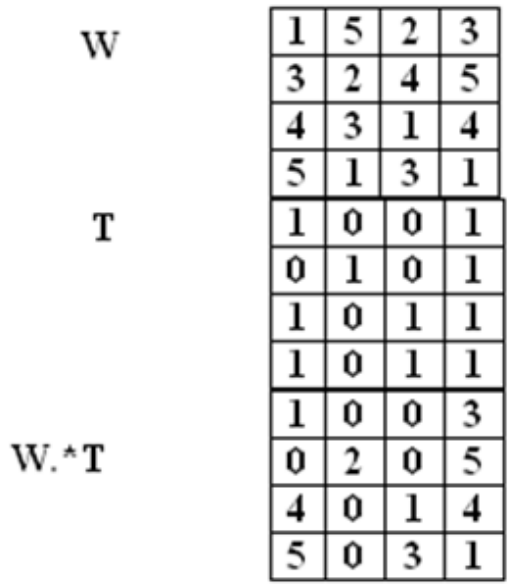


Figure 1. *Exemplary placement of the weight coefficients*

Figure 1 shows that the indicator C of the matrix T is 7/10 (K = 7, R = 10), while the use of weight coefficients C decreases to 1/10 (K = 1, R = 10). The number of the weight coefficients from Figure 1 is w = 5, located randomly in weight matrix W. It is obvious that the non-conflict request commutation improves at times (C decreases at times). The price of this is a decrease of the performance at times. The aim is a reasonable compromise, in which C is minimal at a relatively small number of weight factors [Kolchakov and Monov, 2014a].

From the graphics of Figure 2 it is visible that for N = 4, the use of two weights leads to a seven-fold decrease of C. When N = 32 even in ten weights C is not changed and is in non-sensitive zone. In case of N = 2, C is permanently zero for six of weights [Kolchakov and Monov, 2014a]. In order to avoid the non-sensitive zone it is created an algorithm based on the approach for traffic management with weight factors in Crossbar commutator.

For a matrix of T requests, it is applied a family of weight matrices W. In each weight matrix it is determined value of C. If C = 0 iterations stop, because it is found a weight matrix that guarantees non-conflict schedule [Kolchakov and Monov, 2014b].

A software model of the algorithm is synthesized written in Matlab 6.5 language. Computer configuration Dell OPTIPLEX 745 (Core 2 Duo E6400 2,13GHz, RAM 2048) is used [Kolchakov and Monov, 2014b]. The study is done for N = const., T = const. and a family of weight matrices (W) with a different number of weights (w), selected in random law. The results are presented graphically on Figure 3 [Kolchakov and Monov, 2014b].

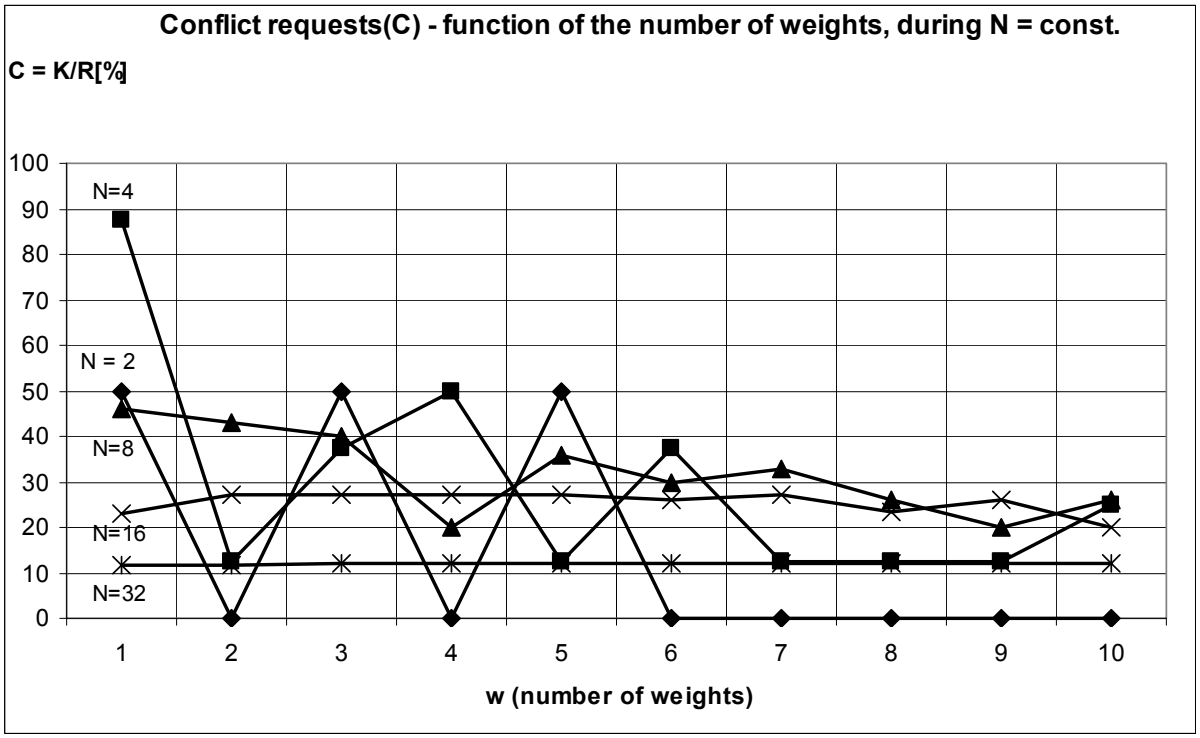


Figure 2. Conflict requests (C) – function of the number of weights, during N = const

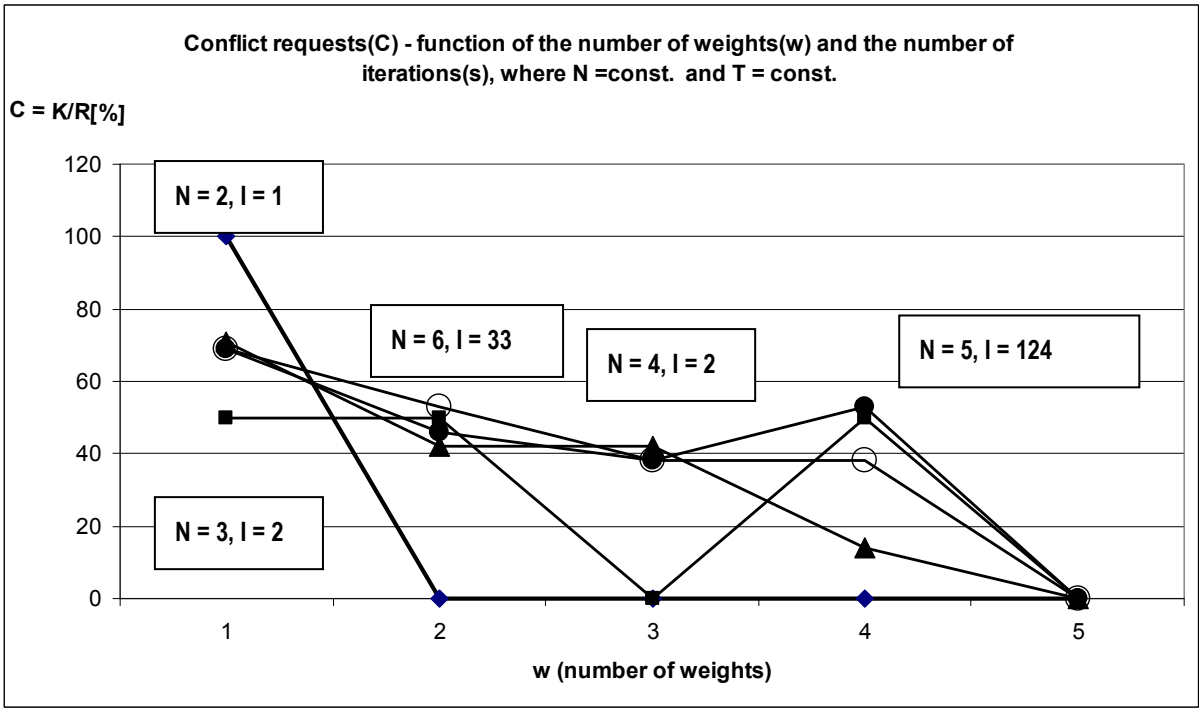


Figure 3. Conflict requests (C) – function of the number of weights and the number of iterations (I), where N = const. and T = const

From the results on Figure 3 it is visible that when N = 2 it is sufficient one iteration to achieve C = 0 when w = 2, while for N = 3, C is set to zero at w = 3 for two iterations [Kolchakov and Monov, 2014b].

When N = 5 only for w = 5 and 124 iterations C becomes zero.

It is important to note, that the algorithm ensures always a final solution and the non-sensitive zone is avoided. For a given number of iterations, or increase of the number of weights a non-conflict schedule (C = 0) is achieved [Kolchakov and Monov, 2014b].

Adaptive Algorithm for Traffic Management in Crossbar Commutator Description

An essential element in the adaptive algorithm is the requests detector in the matrix of connections. The detector finds requests by scanning the matrix of connections and assigns different weight factor of each request. The weights manage commutator switches so that the high weight corresponds to a high priority. The strategy is to increase the weights from left to right and from top to bottom in the weight matrix.

It is important to note that the selected approach leads to a non-conflict schedule (C = 0). Figure 4 shows the result of the operation of the adaptive algorithm for management (AAM).

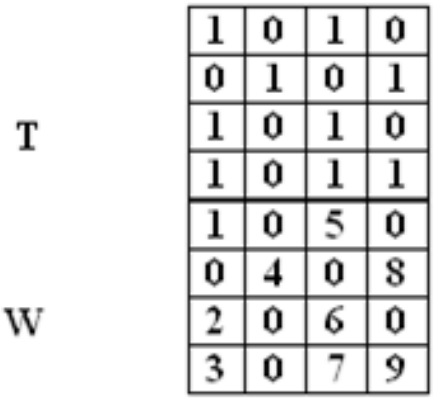


Figure 4. *The result of the operation of the adaptive algorithm*

From Figure 4 it is seen that the number of weights is equal to the number of requests which is quite wasteful and leads to delay in the implementation. Obviously it is needed to optimize the process in terms of the number of weights.

Using the knowledge that diagonally located requests are non-conflict to one another could reduce the number of weights. It is enough the requests detector to group the requests by diagonals and to assign one and the same weight within the diagonal.

Figure 5 shows the result of an optimized version of the adaptive algorithm (AAMO).

From Figure 5 it is clear that in the optimized variant of the adaptive algorithm for one and the same number of requests with one and the same location in the connections matrix T, the number of the weights decreases from 9 to 5.

In the adaptive algorithm (AAM) in the limiting case where the maximum number of requests is equal to NxN the number of weights is equal to the NxN also, while in the optimized version of the adaptive algorithm (AAMO) the number of weights is 2N-1 only.

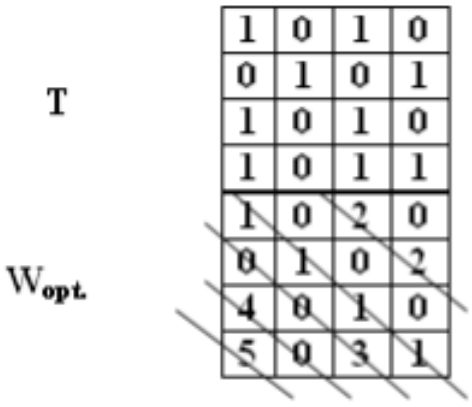


Figure 5. *The result of an optimized version of the adaptive algorithm*

We have synthesized software models of the adaptive algorithm (SMAAM) and its optimized version (SMAAMO) which is written in the language of Matlab 6.5. Computer configuration Dell OPTIPLEX 745 (Core 2 Duo E6400 2,13GHz, RAM 2048) is used for simulations.

SMAAM and SMAAMO are tested at T = const in terms of the required number of weights to achieve a non-conflict schedule (C = 0). The results of this test are shown in Table 1 and Figure 6.

Table 1. *Required number of weights (w) for C = 0 at T = const*

N	**w** by **SMAAM**	**w** by **SMAAMO**
4	9	6
8	26	12
16	133	28
32	502	63

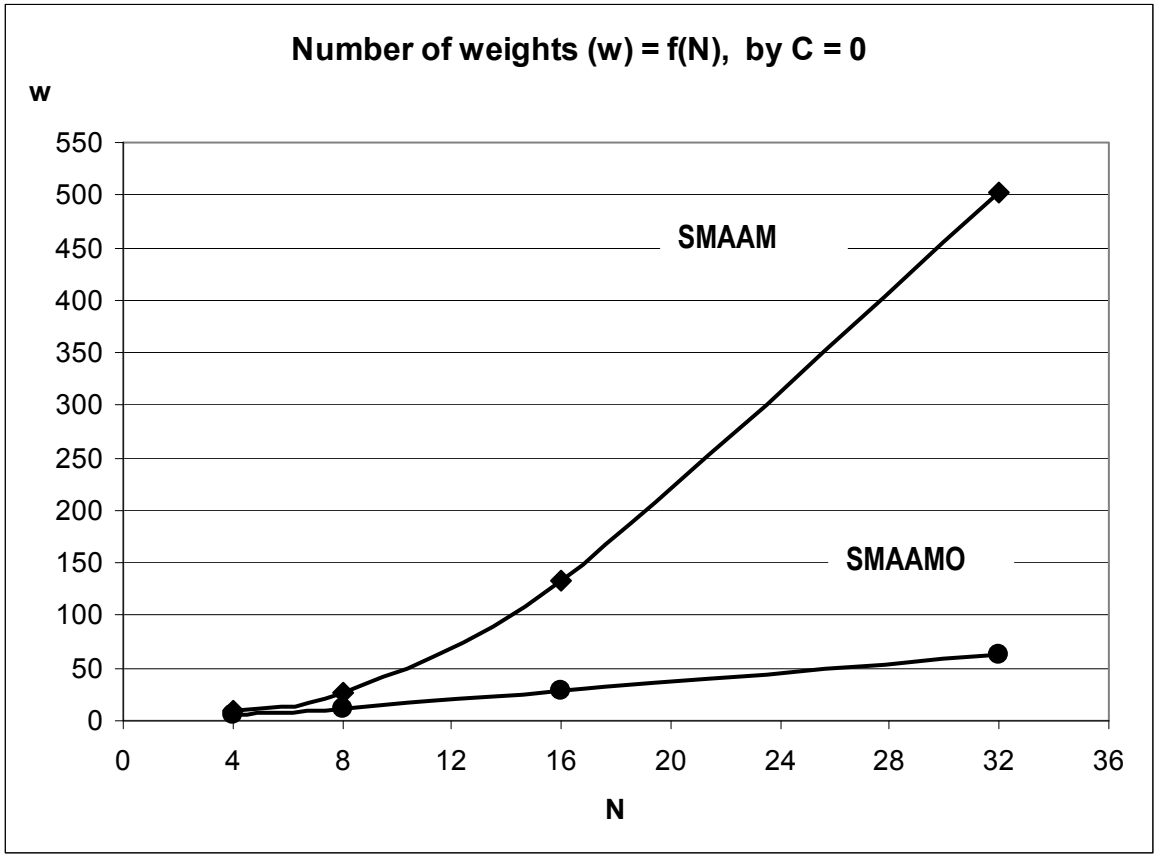


Figure 6. *The required number of weights to achieve non-conflict schedule*

Conclusion

From the performed tests it is seen that the adaptive algorithm (AAM) for the traffic management in Crossbar commutator with weight factors can be used for large size of the matrix of connections. The disadvantage is the relatively large number of weight factors.

In the optimized version of the adaptive algorithm (AAMO) that disadvantage is avoided. When N = 32, the number of weights in the optimized version is less than seven times, compared to the baseline adaptive algorithm.

Bibliography

[D. Kim, K. Lee and H. Yoo, 2005] "A Reconfigurable Crossbar Switch with Adaptive Bandwidth Control for Networks-on-Chip", IEEE International Symposium on Circuits and Systems, 2005

[Kolchakov and Monov, 2013] Kolchakov K, Monov V. An Algorithm for Non – Conflict Schedule with Diagonal Activation of Joint Sub Matrices, 17-th International Conference on "DISTRIBUTED COMPUTER AND COMMUNICATION NETWORKS (DCCN-2013):"CONTROL, COMPUTATION, COMMUNICATIONS, Moscow, October 07-10, 2013, pp. 180-187, ISBN 978-5-94836-366-0.

[Kolchakov and Monov, 2014a] Kolchakov K, Monov V. "Management Approach by Weight Coefficients of the Traffic in Crossbar Commutator" Proceedings of the International Conference AUTOMATICS AND INFORMATICS`14, John Atanasoff Society of Automatics and Informatics, Bulgaria, Sofia 01.10-03.10.2014, pp. I-159 – I-161, Proceedings ISSN 1313-1850, CD ISSN 1313-1869.

[Kolchakov and Monov, 2014b] Kolchakov K, Monov V. "Algorithm for Management by weight Coefficients of the Traffic in Crossbar Commutator", Proceedings of the International Conference ROBOTICS, AUTOMATION AND MECHATRONICS – RAM 2014, 5-7 November 2014, Sofia , Bulgaria pp. .

[Kolchakov and Taschev, 2012] Kolchakov K., Taschev T. "An Algorithm of Non-Conflict Schedule with joint Diagonals activation of Connectivity Matrix", Proceedings of the International Conference on Computer Systems-CompSysTech'12, 22-23 June 2012, Ruse, Bulgaria, ACM PRESS, ICPS, Vol.630, pp.245-250, ISBN 978-1-4503-1193-9.

[Kolchakov K., 2009] Kolchakov K., Software Models with Sparse Mask Matrixes for Non-Conflict Schedule obtaining in a Switching Node, Proceedings of the International Conference "Distributed Computer and communication networks DCCN 2009", 05-09 October 2009 Sofia, Bulgaria. R&D Company "Information and Networking technologies", Moscow, Russia, pp. 121-126. ISBN 978-5-9901871-1-5

[Kolchakov K., 2010] Kolchakov K., "An Approach for Performance Improvement of Class of Algorithms for Synthesis of Non-conflict Schedule in the Switch Nodes", Proceedings of the 11th International Conference on Computer Systems and Technologies, CompSysTech'10, 17-18 June 2010, Sofia, Bulgaria, pp.235-239, ISBN 978-1-4503-0243-2.

[Kolchakov K., 2011] Kolchakov K., "An Algorithm Synthesis of Non-Conflict Schedule by Diagonal Connectivity Matrix Activation" Proceedings of the International Conference AUTOMATICS AND INFORMATICS`11, John

Atanasoff Society of Automatics and Informatics, Bulgaria, Sofia 03.10-07.10.2011., pp. B-247 – B251, Proceedings ISSN 1313-1850, CD ISSN 1313-1869.

[Kolchakov K., 2011] Kolchakov, K., Synthesis of an Optimized Non-conflict Schedule Accounting the Direction of Messages Transfer in a Communication Node. Cybernetics and Information Technologies, Bulgarian Academy of Sciences, Vol. 4, No 2, 2004, pp. 88-95, Sofia.

[M.Schaller and K. Svozil, 1996] "Automaton Logic", International Journal of Theoretical Physics, Vol.35, No. 5, pp. 911-940, 1996, Plenum Publishing Corporation-Springer.

[P. Wanjari and A. Chonhari, 2011] "Implementation of 4x4 crossbar switch for Network Processor", International Journal of Emerging Technology and Advanced Engineering, Website: www.ijetae.com (ISSN 2250 – 2459, Volume 1, Issue 2, and December, 2011)

[Tashev T., 2010] Tashev T. Computering simulation of schedule algorithm for high performance packet switch node modeled by the apparatus of generalized nets. Proceedings of the 11th International Conference CompSysTech' 2010, 17-18 June 2010, Sofia, Bulgaria, ACM ICPS, Vol.471, pp.240-245

Authors' Information

Kiril Kolchakov *- Department "Modelling and Optimization", Institute of Information and Communication Technologies – Bulgarian Academy of Sciences, "Acad. G. Bonchev" bl. 2, Sofia 1113, Bulgaria; e-mail: kkolchakov@iit.bas.bg*

Major Fields of Scientific Research: Distributed Information Systems Design, Methods and tools for net models researches.

Vladimir Monov - Department *"Modeling and Optimization", Institute of Information and Communication Technologies – Bulgarian Academy of Sciences, "Acad. G. Bonchev" bl. 2, Sofia 1113, Bulgaria; e-mail: vmonov@iit.bas.bg*

Major Fields of Scientific Research: Distributed Information Systems Design.

AN APPROACH TO BEHAVIORAL SOFTWARE MODELS ANALYTICAL REPRESENTATION

Elena Chebanyuk

Abstract: *According to standard UML 2.5 Behavioral Software Models (BSMs) are UML diagrams that represent software behavior [UML 2.5, 2012]. An approach of BSM analytical representation is proposed in this paper. Such BSMs as Use Case Diagrams (UCDs), Collaboration Diagrams (CDs) and Sequence Diagrams (SDs) are considered at this paper. In order to design this approach objects and operations that are both common and specific for every type of considering UML diagrams are defined.*

In order to prepare general analytical representation of BSM it is proposed to form two tuples containing objects and operations. Thus, BSM is represented as a Cartesian product of these two tuples. Then analytical representation of conditional operation is proposed.

The peculiarity of the represented approach is that unified analytical representation of some operation for all types of considering BSMs is used. Such representation facilitates performing many operations in Model-Driven Architecture (MDA) area.

Literature review shows that researches of BSMs mostly focused on preparation of an analytical representation for parallel processes [Hu, 2014], [Dove, 2014] or designing transformation technics [Rhazali, 2014], [Romera ,2014], [FUML, 2013], [Wang, 2014], [Bonaris, 2014] and ect.. But conditional operations are important constituents of any complex algorithms of processes. Existing technics of conditional operations description, for example Object Constraint Language (OCL) [OCL 2.4, 2014], do not contain detailed notation for representation of any condition with given precision level. The notation for formal analytical representation of conditional operation is represented in this paper. It is shown that any complex conditions can be described in terms of the proposed notation. In order to prove this thesis different conditional blocks are designed from Collaboration Diagrams. Also the analytical representation of conditional block is obtained from other types of considering BSM.

Also in the point further research the general idea of framework for the analytical description of BSM is proposed.

In conclusion advantages of implementing of the proposed approach to perform different activities in software development lifecycle are formulated.

Keywords: *Analytical representation, UML diagrams, Set-theory, Object Constraint Languages (OCL), Model-Driven Architecture (MDA).*

ACM Classification Keywords: *D.2.2 - Design Tools and Techniques, D.2.11 Software Architecture*

Introduction

Today software development lifecycles that correspond to Agile methodology can better meet customer requirements than Rational Unified Approach (RUP). Involving Model-Driven Development (MDD) approach into software development lifecycles helps to raise an effectiveness of different processes [Swithinbank, 2005]. Software models, which are represented as UML diagrams, are central artifacts of MDD approach [UML 2.5, 2012]. The classification of UML models is represented in the paper [Chebanyuk, 2014] In MDD and MDA approaches BSM are Computation Independent Models (CIMs). Computation Independent Models are metamodels which contain basic information about software requirements and functionality [Chebanyuk, 2014]. Effective processing of CIM models allows obtaining qualitative initial information to design the rest of software artifacts. Preparing an analytical representation of BSM allows implementing a background for many MDA operation. These operations are model comparing, refactoring, reusing, merging and so on. Such operations are performed in different software development lifecycle activities, for example requirement verification and validation, algorithms analyzing and refinement, interface designing, testing, creating and modifying documentation and other.

Related papers

Analytical representation of BSM is an initial step for different model processing operations. These operations can be divided on two categories, namely software model transformations and processing.

Consider papers that are devoted to BSMs transformation in MDA sphere

Paper [Rhazali, 2014] proposes a method for transformation CIMs to Platform Independent Models (PIMs). The first step of this method is refining BSM, namely UC or CD, according to specific rules. Every requirement is associated with some business process. Then some use cases are mapped into classes. However this method doesn't contain defined rules for matching description of requirements and model elements to some business processes. This fact makes difficult usage both mathematical apparatus and formal languages for analyzing requirement specification. Also formal modeling requires a formal language. A formal language, unlike a natural language, is one that operates with explicitly defined rules.

In the paper [Wang, 2014] approaches that are used for software models transformation are systematized. The next approaches are considered: Template-Based Approaches, Target-Structure-Driven Approaches, Graph-Transformation-Based Approach, Relational Approaches and Approach with models serializing. The short description of these approaches is represented below.

Template-Based Approaches In this approaches, templates consisting of text in the target language include meta-code tags to access information from the source model. In the transformation process,

these tags will be interpreted and eventually replaced by code representing the corresponding parts of the source. These approaches usually cover model to code generation operations.

Target-Structure-Driven Approaches The basic metaphor is the idea of copying model elements from the source to the target. This kind of approaches was developed in the context of certain kinds of applications such as generating Enterprise Java Beans (EJB) implementations and database schemas from UML models.

Graph-Transformation-Based Approaches This category of model transformation approaches draws on the theoretical work on graph transformation. In particular, these approaches operate on typed, attributed, labeled graphs, which is a kind of graphs specifically designed to represent UML-like models. This kind of approaches is inspired by heavily theoretical work in graph transformations, and it is powerful and declarative, but also the most complex ones.

Relational Approaches This kind of approaches uses the mathematical concept of relations to specify how source and target models are linked. Relations are declarative but may be given execution semantics. It seems to strike a well balance between flexibility and declarative expression. Transformation Implemented using Extensible Stylesheet Language Transformations (XSLT). Models can be serialized as Extensible Markup Language (XML) using the XML Metadata Interchange (XMI), and implementing model transformations using XSLT. Most of the approaches given above focus on providing a concrete solution for the transformation from PIMs to Platform Specific Models (PSMs), and there is little research on the definition principles for mapping rules as well as a basic theory to validate the mapping rules between such models. The research about natural language translation by machine shows that the prerequisite of correct transformation between different languages is the same or similar characteristics of semantics expression within the source and the target.

Semantic-element-based Defining Approach for Model A model mapping approach based on semantic consistency was proposed by abstractly analyzing the characteristic of syntax and semantics of modeling languages. Abstract target semantic model must be firstly constructed through an in-depth analysis of target platform. Then, based on the idea of elements in source semantic domain being reconstructed in the target semantic domain, mapping relations from source model to target model are created via abstract target semantic model. The formalization of model transformation approach which uses patterns is proposed and is represented by the following: A pattern can be defined as a 3-tuple:

$$P =< C, A, SR > \quad (1)$$

where C= {c| c is a conceptual element in PIM or PSM}, and A= {a| a is a relevant attributes of the conation operations are performed}.

The approaches, proposed in the paper [Wang, 2014], allow estimating the effectiveness of transformation operations from the different points of view. They can provide a general framework for

modifying and designing new model mapping technics when methods of software models transformation are designed.

Solving the task of formal processes description helps to prepare initial information for different transformation tasks. Examples of these task are: transformation business rules to class diagram [Bolaris, 2014], code generation from PSM models [FUML, 2013] and evaluation of system design [Romero, 2014].

Complex description of transformation approach from business rules to class diagrams is represented in the paper [Bonais, 2014]. Authors proposed description of all steps to perform this task. Formal models for entire and resulting information are also represented. Also constrains for initial and resulting models are proposed. Further rules for mapping process to class diagram constituents are formulated. This approach can be used for software requirement elicitation. One can use business rules of problem domain to define whether requirement specification corresponds to business needs. Also this approach can be used both for UML profile elicitation and class diagram refinement. Formal model for business rules representation contains vocabulary. Vocabulary matches facts to some objects or entities of problem domain. Second component of the proposed model is a set of rules. Every business rule is represented as the tuple

$$br = (statement, factType, \mathrm{mod}al, quatifier1, quantifier2) \quad (2)$$

where: *statement* - is the statement of the business rule. It is built by combining fact types and keywords;

factType - is the fact type that is used to build the business rule statement;

modal - is the modal keyword that is included in the business rule statement;

quantifier1 - is the first quantification keyword that is included in the business rule statement. It always

belongs to the first noun concept of the fact type;

quantifier2 - is the second quantification keyword that is employed in the business rule statement. It

always belongs to the second noun concept of the fact type.

Representation of business rules (2) can be used for formal analysis of functional requirements, algorithms, processes, comparing and refinement requirements specifications. Using this notation one can perform different operations of analyzing text information by means of formal models processing.

Standard FUML describes an intermediated framework between PSM model and code [FUML, 2013]. Target platform language is Java. Elements of a class are described as a tuple of properties, relationships and operations. Other tuples are association classes, interfaces and dependencies. Then

notation for description of class behavior is proposed. Also templates of code for every variant of classes are represented.

Also the collection of code templates is prepared. Classes are connected as constituents of design patterns. Thus, FUML activities are the next:

- Define behavior features of component.
- Represent the formal description of class.
- Compare this description with analytical representation of classes.
- Define the closest analytical representation of class and represent the corresponding template of code.

Authors [Romero, 2014] proposed a FUML interpreter. This interpreter verifies BSM. In order to perform this task authors presented the relationships between four components, namely abstract syntax, execution model, semantic domain, and base semantics.

The execution model is an interpreter written in FUML objects and object nodes, which are interconnected by unary and binary relations. However, the base semantics for designing an executable code is represented. However number of building blocks that are designed for generation of code fragments is limited. And even small differences in processes can be causes of big differences in codes. Thus, in order to obtain executable code, reflecting the peculiarities of processes it is necessary to do the following:

- propose more precise notation of the abstract syntax;
- increase number of basic semantic elements that reflect differences in code;
- add to the base of code templates by new fragments of code;
- propose a technique of design patterns reusing when their number will be increased;
- include specific attributes to the description of processes behavioral features. The aim of these attributes is to provide more precise mapping of model processes to code templates.

Authors [Hu, 2014] says that many efforts were made to make UML executable by transforming single diagram to executable model such as Colored Petri Nets (CPN), however, approach like this could not provide more systematic and intuitive simulation of the entire system. Therefore it was proposed to use the Discrete Event System Specification (DEVS) as the target formalism and transform UML Diagrams to executable models.

Firstly two types of models for considering parallel execution of processes is proposed. The atomic model is the irreducible model definition that specifies the behavior for any modeled entity. The coupled model represents the composition of several atomic and coupled models connected by explicit couplings.

An atomic model M and a coupled model N are defined by the following equations:

$$M =< IP, OP, X, S, Y, \vartheta\text{int}, \vartheta ext, \vartheta con.\lambda, ta > \quad (3)$$

$$N =< IP, OP, X, Y, D, EIC, EOC, IC > \quad (4)$$

where:

- *S*- is the state space;
- *IP, OP* - are the set of input and output ports;
- *X, Y* - are the set of Inputs/Outputs, which are basically lists of port-value pairs;
- X = {(p,v) | p ∈ IP, v ∈ Xp }, Y = {(p,v) | p ∈ OP, v ∈ Yp }, where Xp and Yp are input/output values on port p;
- ϑint :S → S is the internal transition function;
- ϑext :Q × Xb → S is the external transition function, where Q = {(s,e) s ∈ S, 0 ≤ e ≤ ta(s)} is the total state set,
- ϑcon : S × Xb → S is the confluent transition function, which decides the order between ϑint and ϑext
- in cases of collision between simultaneous external and internal events;
- λ : S → Yb is the output function, where Yb is a set of bags composed of elements in Y;
- ta(s): S → R0+ ∪∞ is the time advance function which decide how much time the system stays at the current state in the absence of external events.

Extended DEVS modeling language has the following components: Models for Separation of Message Processing From State Transition and the abstract syntax. In terms of abstract syntax software models are divided on three elements, namely the Entity, the Atomic and the Coupled. Also functions for more precise description of atomic models are presented. For code generation operations special language Xpand is used. Then executable codes are obtained through Xtext framework because Xtext is seamlessly integrated with the Eclipse Java framework by code generator with Xtend. Also the proposed approach can be used to refine class and component diagrams. Entity “component” of model can be mapped to class in atomic model representation. And Component Diagram provides structural information about a coupled model including components in coupled model and couplings between them.

However extended DEVS language covers just a sphere of parallel execution of some processes. Such operations as conditional statements, cycle constructions, collection processing and other are not described by this language.

Also set-theory serves for representation of different constrains of software models. For example using OCL one can define a set of model constrains that reflects specific needs of a problem domain.

A UML diagram, such as a class diagram, is typically not refined enough to provide all the relevant aspects of a specification. There is, among other things, a need to describe additional constraints about

the objects in the model. Such constraints are often described in natural language. Practice has shown that this will always result in ambiguities. In order to write unambiguous constraints, so-called formal languages have been developed. The disadvantage of traditional formal languages is that they are usable to persons with a strong mathematical background, but difficult for the average business or system modeler to use OCL [OCL 2.4, 2014]. .

OCL has been developed to fill this gap. It is a formal language that remains easy to read and write. It has been developed as a business modeling language within the IBM Insurance division, and has its roots in the Syntropy method [OCL 2.4, 2014].

OCL has very flexible semantic. Using text notation one can represent some meaning of variables and set of the restrictions. These restrictions are invariants, pre and post conditions, initial and derived values and others. Flexibility of OCL notation allows including new elements to meet specific requirements of representation of peculiarities of business processes.

One of direction of these researches to precise the types of object interaction by means of an association link between object. Authors [Dove, 2014] proposed to extend the OCL notation to better express UML qualified association. A qualified association is an association that allows one to restrict the objects relations using a key

called a qualifier. Additional constrains are added. They are formulated by means of set theory and predicate logic and supplements UML diagrams by providing expressions.

The key idea of the proposed approach is to make a qualified association a first class entity in OCL by exposing it as a map and allowing one to query and manipulate the map directly. For this a small extension to the OCL notation was proposed to denote the qualified association map and introduce a new collection library class to model a map. The collection of map specific operations was proposed. Pre and post conditions, rules for defining collection processing, equality and defining operation are defined. Listed operations can help to define relations between objects more precise.

Also researches are performed for solving the task of conceptualization the different facts of component model in Component Based Software Engineering (CBSE). This approach facilitates modeling of components using different schemas in software system. One of the proposed solutions is to use Z-semantic based conceptual model of component, called Z-Formal Specification of Component Model (ZFSCM). Is describes formal description of components and interconnections between them [Banerjee, 2014].

Z uses not only mathematical notation to describe properties of information system without constraining the way in which these properties are achieved, it also uses the schema to define these properties. They describe what the system must do instead of saying how it is to be done. This abstraction makes Z useful in the process of developing a system. Z-specification can serve as a single, reliable reference

point for those who investigate the customer's needs, test results and write instruction for the system. It makes easier to write mathematical description of complex dynamic software systems. The descriptions are usually smaller and simpler than any programming language can provide. They should contain a mixture of formal and informal parts. Z is based on the standard mathematical notation using mathematical data types to model the data in a system. It also used in axiomatic set theory, first-order predictive logic, lambda calculus. Another thing is that it has a way of decomposing a specification into small piece called schemas. After splitting the specification into schemas, it can present it piece by piece. Every piece is connected with formal mathematics. All expressions in Z notation are typed, thereby avoiding few of the paradoxes of set theory. Z contains a standard mathematical toolkit of commonly used mathematical functions and predicates, called catalog. Bur the task of designing flexible approach to describe BSM processes is remains actual.

Authors [Banerjee, 2014] proposed graphical notation for description of main elements os component diagram, namely service, interface, class and components, and relationships between them. Also they represented text-based description for type-checking. In order to design a component model it is proposed every requirement in specification match with service, then service join to components, after that, the model of components is designed. During design process two types of interfaces are defined, namely required and used. This model can be used for such model processing operations as components reuse and making changes to the components. This model can be complicated by other elements to describe different features of components and service that are represented in OMG standards for description of Service oriented modeling languages [SOAML, 2010] and UML components diagram [UML 2.5, 2012].

Task and challenges

The task is to design the approach for an analytical representation of BSM. In order to perform this task it is necessary to do the following:

- consider a general form of the analytical representation for three kinds of BSMs that are commonly used in software development activities, namely UCDs, CDs and SDs;
- define both common and specific objects and operations that are used in UML notation for every considering type of BSM;
- discover key features of operations for representing an analytical description of BSM;
- introduce the denotation for the analytical representation of conditional operation;
- prove that the representation of conditional operations is unified for different types of BSM.

Challenges are: the analytical representation of BSM should satisfy to the following conditions:

- contain such a notation that allows designing format of file for saving information about BSM. This notation should allow modifying a structure of file easily when BSMs are changed;

- be easy combined with technics [Hu, 2014], [Chebanyuk, 2014] and standards [FUML, 2013] that allow to store and visualize information about software model, including of sequence of its processes and sub process;
- consider all entire information for performing of operations processing models such as comparing, refactoring, reusing and others;
- allow to formalize representation of BSM for solving the MDA task of transformation between BSM of different types.

The proposed approach

The task of every BSM is to represent processes. Every BSM has its own notation. This notation allows representing processes with given degree of details.

For the analytical of BSM the Set-Theory tool is chosen. It is proposed to represent interconnection between constituents of BSM by means of multiplying two tuples of Cartesian product. One factor of the Cartesian product represents a BSM constituent and another one represents an operation defined in the notation of specific BSM.

Consider such BSMs as UCDs, CDs and SDs. According to UML standard these diagrams are designed using both common and specific elements. For example, the SD notation has more elements and operations to provide detailed representation of processes in comparison with AD or CD.

Forming tuples, containing constituents of different behavioral software models

Table 1 introduces the denotations of constituents in considering BSM according to UML standard.

Table 1. Denotations of BSM constituents

Denotation from the UML standard	Denotation of the proposed approach	Use case	Collaboration	Sequence
1	2	3	4	5
Object	O	-	+	+
1	2	3	4	5

Denotation from the UML standard	Denotation of the proposed approach	Use case	Collaboration	Sequence
Complex object (object with properties defined)	O_c	-	+	-
Actor	Θ	+	+	+
Message	M	+ (precedent is an equivalent of message)	+	+
Collection	Y	-	+	+
Subsystem	S	+	+	+
Multiplicity	N	+	+	-
Waiting for response	I	-	-	+

Using Table 1 tuples of sets from constituents of different BSMs are formed.

Tuple for description of Use Case Diagram constituents is denoted as follows:

$$\mathrm{K} =< \Theta, M, S, N > \tag{5}$$

Tuple for description of Collaboration Diagram constituents is denoted as follows:

$$\mathrm{X} =< O, O_c, \Theta, M, Y, S, N > \tag{6}$$

Tuple for description of Sequence Diagram constituents is denoted as follows:

$$H =< O, \Theta, M, Y, S, I > \tag{7}$$

Forming tuples, containing operations of different behavioral software models

Table 2 introduces the denotations of operations in considering BSM according to UML standard.

Table 2. Denotations of BSM operations

Operation	Denotation of the proposed approach	Use case	Collaboration	Sequence
1	2	3	4	5
Object Creation	Δ	-	+	+
Element of collection processing	E	-	+	+
Waiting for response	Ψ	-	-	+
Representing different types of messages	Π	+ (<<include>> <<extend>> <<generalize>>)	-	+ (Asynchronous call Synchronous call Call of object to itself)
Conditional statement	Ω	+	+	+
1	2	3	4	5
Cycle	Σ	-	+	+
Parallel execution	Ξ	-	+	+

Using Table 2 tuples of sets from constituents of different BSM notations are formed.

Tuple for description of Use Case Diagram operations is denoted as follows:

$$\Phi =< \Pi, \Omega > \tag{8}$$

Tuple for description of Collaboration Diagram operations is denoted as follows:

$$\Gamma =< \Delta, E, \Omega, \Sigma, \Xi > \tag{9}$$

Tuple for description of Sequence Diagram operations is denoted as follows:

$$\Lambda =< \Delta, \mathrm{E}, \Psi, \Pi, \Omega, \Sigma, \Xi > \tag{10}$$

Preparing an analytical representation of behavioral software models

General form of BSM representation by means of the Cartesian product is denoted as follows:

$$\begin{cases} \mathrm{Z}^{*} \subseteq Con \times Oper \\ Con = \begin{cases} \mathrm{K} \\ \mathrm{X} \\ H \end{cases} \\ Oper = \begin{cases} \Phi \ then \ Con = K \ and \ * = UC \\ \Gamma \ then \ Con = X \ and \ * = C \\ \Lambda \ then \ Con = H \ and \ * = S \end{cases} \end{cases} \tag{11}$$

where Z – is the representation of whole BSM. Its upper index * shows the type of BSM, namely

- * – UC – Use Case Diagram;
- * – C – Collaboration Diagram;
- * – S – Sequence Diagram.
- Oper – the set of BSM operations
- Con - the set of BSM constituents.

Consider an example of Use case Diagram analytical representation according to the proposed approach

$$\mathrm{Z}^{UC} \subseteq K \times \Phi \tag{12}$$

In order to represent an analytical description of operations specific “building blocks” can be designed. These “building blocks” should contain key features of operation. The examples of analytical representation of UCD operations is introduced in the paper [Chebanyuk, 2014].

For example, consider an analytical representation of conditional statements for every type of BSM. Analyzing Table 2 one can see that conditional operations are met in three types of BSMs. From the other hand literature review shows that there is no technics for the analytical representation of complex conditional operation with given level of precision.

Consider a procedure of preparing the analytical representation of conditional operation. This representation should satisfy to the main feature of the proposed approach, namely, to be unified for all types of BSMs.

Designing the unified analytical representation of conditional statement for the all types of considering behavioral software models

The first step of this procedure is to define key features of this operation. Table 3 represents key features for the analytical description of conditional operations, parallel statements and also cycle operations.

Table 3. Denotations of some behavioral software models operations

Conditional statement	Parallel execution of operation	Cycle
Object from which brunching is started (root object) Condition (can be both simple or complex) Action or actions that are executed when condition is true (Optional) Action or actions that are executed when condition is false (Optional) Action or actions that are executed when all conditions are false (Optional) object, which meets all conditional brunches	Condition when parallel processes are started. See description of conditional statements. Action or actions that are executed when every thread of the parallel process is executed A marker (may be process or an object) from which parallel execution is started A marker (may be process or an object) showing that parallel execution is ended	Cycle conditions Counter values (optional) Cycle should contain as many analytical blocks as many conditions are in the cycle. See description of conditional statement

Table 3 shows that more complex operations contain conditions as a part of them.

Consider the process of designing building block for conditional statements representation using key features, of its operation that are represented in the Table 3.

Let us assume the following denotations:

c – conditional operation.

root - object from which brunching is started (root object for the Table 3).

ω_i - is a condition with number *i* where *i=1,…,n* and *n* is a number of conditions that are started from the root object.

brunch – object that meets all condition brunches.

An analytical representation of actions which are related to condition ω_i is denoted as follows:

$$< \omega_i > \begin{bmatrix} \omega_i = true(actions\ that\ are\ preformed\ when\ \omega_i = true) \\ \omega_i = false(actions\ that\ are\ preformed\ when\ \omega_i = false) \end{bmatrix} \quad (13)$$

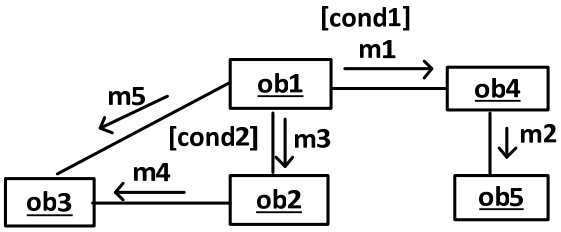
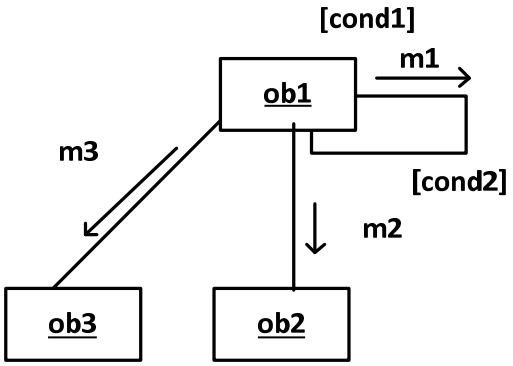
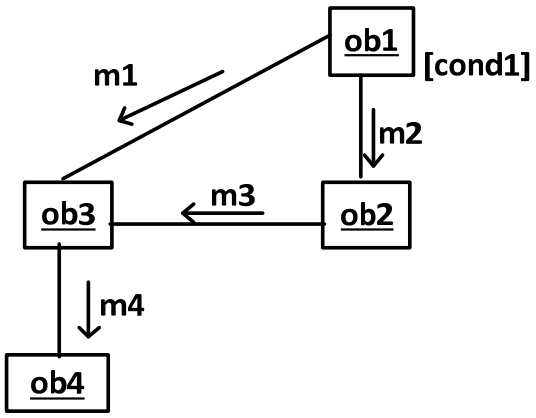
Using the represented denotations introduce a general form of condition operation analytical representation.

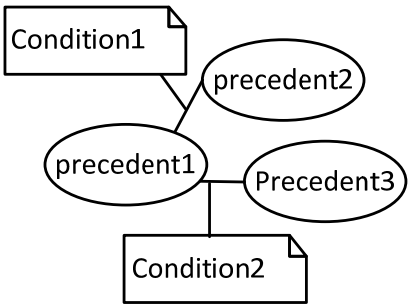
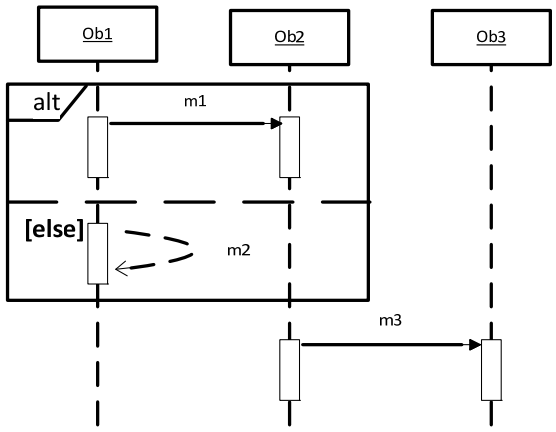
$$c = \begin{cases} root & \\ < \omega_1 > & \begin{bmatrix} \omega_i = true(actions\ that\ are\ preformed\ when\ \omega_1 = true) \\ \omega_i = false(actions\ that\ are\ preformed\ when\ \omega_1 = false) \end{bmatrix} \\ \cdots & \\ \cdots & \\ < \omega_n > & \begin{bmatrix} \omega_n = true(actions\ that\ are\ preformed\ when\ \omega_n = true) \\ \omega_n = false(actions\ that\ are\ preformed\ when\ \omega_n = false) \end{bmatrix} \\ else & [actions\ that\ are\ performed\ when\ all\ conditions\ are\ false] \\ brunch & \end{cases} \quad (14)$$

Analytical representation of conditional operations for different types of behavioral software models

Prove that statement (14) can be applied to three types of BSMs. For this consider different types of processes representation analyzing peculiarities of considered BSMs.

Table 4. Examples of BSMs notation application

BSM fragment including conditional operations	Analytical representation of BSM fragment
Collaboration diagram	
	$Z^c \subseteq \Gamma \times X$ $\Gamma =< \Omega > where\, \Omega = \{\omega_1, \omega_2 \mid \omega_1 = cond_1, \omega_2 = cond_2\}$ $X =< O, M > where\, O = \{ob_1, ob_2.ob_3, ob_4, ob_5\}$ $M = \{m_1, m_2, m_3, m_4, m_5\}$ $c = \begin{cases} ob_1 \\ < cond_1 > \\ < cond_2 > \\ else \end{cases} \begin{cases} \\ [cond_1 = true(< m_1, ob_4 >, < m_2, ob_5 >)] \\ [cond_2 = true(< m_3, ob_2 >, < m_4, ob_3 >)] \\ [< m_5, ob_3 >] \end{cases}$ $Z^c \subseteq < c >$
	$Z^c \subseteq \Gamma \times X$ $\Gamma =< \Omega > where\ \Omega = \{\omega_1 = cond_1, \omega_2 = cond_2\}$ $X =< O, M > where\, O = \{ob_1, ob_2.ob_3\}$ $M = \{m_1, m_2, m_3\}$ $c = \begin{cases} ob_1 \\ < cond_1 > \\ < cond_2 > \\ else \end{cases} \begin{cases} \\ [cond_1 = true(< m_1, ob_1 >)] \\ [cond_2 = true(< m_2, ob_2 >)] \\ < m_3, ob_3 > \end{cases}$ $Z^c \subseteq < c >$
	$Z^c \subseteq \Gamma \times X$ $\Gamma =< \Omega > where\ \Omega = \{\omega_1 \mid \omega_1 = cond_1\}$ $X =< O, M > where\, O = \{ob_1, ob_2.ob_3, ob_4\}$ $M = \{m_1, m_2, m_3, m_4\}$ $c = \begin{cases} ob_1 \\ < cond_1 > \\ \\ < (m_1 m_3), ob_3 > \end{cases} \begin{cases} \\ [cond_1 = true(< m_2, ob_2 >, < m_3, ob_3 >)] \\ \\ \end{cases}$ $Z^c \subseteq < c >, < m_4, ob_4 >$

BSM fragment including conditional operations	Analytical representation of BSM fragment
Use Case diagram	
	$Z^{UC} \subseteq K \times \Phi$ $K =< \Omega >, \Omega = \{\omega_1, \omega_2 \mid \omega_1 = Condition1, \omega_2 = Condition2\}$ $\Phi =< M >, M = \{m_1, m_2, m_3 \mid m_1 = precedent1,$ $m_2 = precedent2, m_3 = precedent3\}$ $c = \begin{cases} m_1 \\ < \omega_1 > \left[\omega_1 = true(< m_2 >)\right] \\ < \omega_2 > \left[\omega_2 = true(< m_3 >)\right] \end{cases}$ $Z^{UC} \subseteq < c >$
Sequence diagram	
	$Z^{s} \subseteq \Lambda \times H$ $\Lambda =< \Omega > where \ \Omega = \{\omega_1 \mid \omega_1 = alt\}$ $H =< O, M > where \ O = \{ob_1, ob_2, ob_3\}, M = \{m1, m2, m3\}$ $c = \begin{cases} ob_1 \\ < \omega_1 > \left[\omega_1 = true(< m_1, ob_2 >)\right] \\ \left[\omega_1 = false(< m_1, ob_1 >)\right] \end{cases}$ $Z^{s} \subseteq < c >, < m_3, ob_3 >$

Using key features of conditional operation and (14) it is proved that the proposed approach allows to obtain unified analytical representation for the same operation in different types of BSM. Such representation facilitates many operations in MDA sphere and helps to archive an analytical apparatus for precise processing of software models. Thus, in turn, helps to realize many MDD activities more effectively.

Conclusion

The general idea of the approach for the analytical representation of BSM is proposed in this article. Firstly processes are decomposed on operations. Table 2 illustrates types of operations that are typically for different BSM according to UML standard. For the analytical representation of these operations the concept of “building blocks” is proposed. One “building block” corresponds to one type of operation. The detailed analytical representation of conditional operations is proposed. This approach proposes unified notation for an analytical representation of operations in different BSM.

Using this approach one can represent processes with given level of precision and number of details. It provides effective analysis of requirements, test scenarios generation, algorithms elicitation and performing other activities in software development lifecycle that require information about processes. Manipulations with “building blocks” lets to skip, combine or represent some operation with additional details allows focusing just on important features of processes for analysis. For example, when conditional block is used, some number of conditions can be added or removed.

Unified representation of operations allows implementing fast and precise transformations between BSM of considering types. Also integrating of three components, namely: approaches, proposed in paper [Bonais, 2014], methods of transformation CIM to PIM models [Rhazali, 2014] and analytical representation of BSM objects allows to design different transformation technics. Their purpose is to solve the important MDA task, namely, obtaining precise information when resulting BSM contain more detailed information about processes in comparison with initial. Also the advantage of involving unified analytical representation into transformation technics in comparison with approaches, proposed in the paper [Wang, 2014] is that there is no need to design transformation rules or patterns for representation of initial and resulting information.

Also proposed approach lets designing technics for comparing both the same and different types of considering BSMs. Unified representation of operations allows to find precise solution whether the content of different BSMs is matched.

The proposed approach provides a ground for performing such MDA operation as: joining, merging and refactoring software models of the same types. The notation of the proposed approach contains elements, allowing describing both static and behavioral software models (Table 1). It helps to improve notations and technics for different formal specifications. For example, add more complex and precise rules for representation of information systems properties in ZFSCM [Banerjee, 2014].

Using representation of conditional block, proposed in this paper with the OCL language [OCL, 2012] lets remove the limitations when complex conditions are described. Thus, in turn, forms a background for designing UML extensions and profiles considering all peculiarities of an application domain.

Further research

The area of further research is to design a framework for an analytical representation of BSMs. In order to archive this goal it is necessary to do the following:

- define key features of all operations that are specific for considering BSM;
- design analytical representations of these operations;
- propose a notation for storing information about BSMs that includes the sequences of processes;
- design a format of file for storing information about BSMs and technics of visualizing models from these files.

Bibliography

[Banerjee, 2014] Banerjee, P., & Sarkar, A. Z–Specification of Component Based Software. International Journal of Software Engineering and Its Applications, Science & Engineering Research Support Society (SERSC), Australia, 8(1), 2014.

[Bonais, 2014] M. Bonais, K. Nguyen, E. Pardede, W. Rahayu. A formalized transformation process for generating design models from business rules. Proceedings of PACIS 2014 Chengdu, China, 2014 Access mode http://www.sersc.org/journals/IJSH/vol8_no3_2014/8.pdf

[Chebanyuk, 2014] E. Chebanyuk Method of behavioral software models synchronization. International journal “Information models & analyses” Volume 3, Number 2, 2014.

[Dove, 2014] A. Dove, A. Barua, Y. Cheon. Extending OCL to Better Express UML Qualified Associations. In: SEKE'2014: 26th International Conference on Software Engineering and Knowledge Engineering, 2014

[FUML, 2013] OMG standard. Semantic of a foundational subset for executed UML model, 2013 Access mode http://www.omg.org/cgi-bin/doc?formal/2013-08-06.pdf

[Hu, 2014] J. Hu, L. Huang, B. Cao, X. Chang. Extended DEVSML as a Model Transformation Intermediary to Make UML Diagrams Executable. In: Services Computing (SCC), 2014 IEEE International Conference on. IEEE, 2014.

[OCL 2.4, 2014] OMG Standard Object Constraint Language 2.4 Access mode http://www.omg.org/spec/OCL/2.4/

[Rhazali, 2014] Y. Rhazali, Y. Hadi, A. Mouloudi Transformation Method CIM to PIM: From Business Processes Models Defined in BPMN to Use Case and Class Models Defined in UML International Journal of Computer, Information, Systems and Control Engineering Vol:8 No:8, 2014

[SOAML, 2010] Service oriented architectural modeling language Version 1.0.1 OMG standard. Access mode http://www.omg.org/spec/SoaML/1.0.1/PDF

[Swithinbank, 2005] IBM RedBook. Patterns: Model-Driven Development Using IBM Rational Software Architect Access mode http://students.mimuw.edu.pl/~zbyszek/posi/sg247105.pdf

[UML 2.5, 2012] OMG standard. Unified Modeling Language 2.5, 2012 Access mode http://www.omg.org/spec/UML/2.5/Beta1/

[Wang, 2014] L. Wang, Y. Zhang. Semantic-element-based Defining Approach for Model Transformation Rules. International Journal of Smart Home Vol.8, No.3, 2014. Access mode http://dx.doi.org/10.14257/ijsh.2014.8.3.08

Authors' Information

Elena Chebanyuk *– lecturer in National Aviation University, associate professor of software engineering department, Ukraine;*

e-mail: chebanyuk.elena@gmail.com

Major Fields of Scientific Research: Model-Driven Architecture. Domain engineering, Code reuse.

SIMPLE MODEL FOR TRANSMISSION CONTROL PROTOCOL (TCP)

Irma Aslanishvili, Tariel Khvedelidze

***Abstract**: Ad hoc Networks are complex distributed systems that consist of wireless mobile or static nodes that can freely and dynamically self-organize [Namicheishvili et al, 2011]. In this way they form arbitrary, and temporary "Ad hoc" networks topologies, allowing devices to seamlessly interconnect in areas with no pre-existing infrastructure. Introduction of new protocols such as Bluetooth, IEEE 802.11and Hyperlink are making possible the deployment of ad hoc networks for commercial purposes. TCP (Transmission Control Protocol) was designed to provide reliable end-to end delivery of data over unreliable networks. In theory, TCP should be independent of the technology of the underlying infrastructure. In particular, TCP should not care whether the Internet Protocol (IP) is running over wired or wireless connections. In practice, it does matter because most TCP deployments have been carefully designed based on assumptions that are specific to wired networks. Ignoring the properties of wireless transmission can lead to TCP implementations with poor performance. In practice, most TCP deployments have been carefully designed in the context of wired networks. In order to adapt TCP to the ad hoc environment, improvements have been proposed in the literature to help TCP to differentiate between the different types of losses. Indeed, in mobile or static ad hoc networks losses are not always due to network congestion, as it is mostly the case in wired networks. In this paper, we present model: how TCP can be affected by mobility and lower layers protocols.*

***Keywords**: Ad Hoc Networks, protocols, Routing protocols, packet, source node, Relay routing, finite memory, TCP, Simulation Metrics, layers protocols TCP Feedback, TCP-DOOR.*

Introduction

The Transmission Control Protocol (TCP) was designed to provide reliable end-to-end de-livery of data over unreliable networks. In practice, most TCP deployments have been carefully designed in the context of wired networks. Ignoring the properties of wireless ad hoc Networks [Aslanishvili, 2014] can lead to TCP implementations with poor performance. In order to adapt TCP to the ad hoc environment, improvements have been proposed in the literature to help TCP to di errant types of losses. Indeed, in mobile or static ad hoc networks losses are not always due to network congestion, as it is mostly the case in wired networks. In this Chapter, we present an overview of this issue and a detailed discussion of the major factors involved. In particular, we show how TCP can be an acted by mobility and lower

layers protocols. In addition, we survey the main proposals which aim at adapting TCP to mobile and static ad hoc environments.

In ad hoc networks, the principal problem of TCP lies in performing congestion control in case of losses that are not induced by network congestion. Since bit error rates are very low in wired networks, nearly all TCP versions nowadays assume that packet losses are due to congestion. Consequently, when a packet is detected to be lost, either by timeout or by multiple duplicated ACKs, TCP slows down the sending rate by adjusting its congestion window. Unfortunately, wireless networks sure from several types of losses that are not related to congestion, making TCP not adapted to this environment. Numerous en-Han cements and optimizations have been proposed over the last few years to improve TCP performance over one-hop wireless (not necessarily ad hoc) networks.

Simple model for Transmission Control Protocol (TCP)

In the authors report, simulation results on TCP throughput in a static linear multi-hop chain, where IEEE 802.11 protocol is used. In Figure1, we display a multi-hop chain of N nodes. It is expected that, as the number of hops increases, the spatial reuse will also increase. However, simulation results indicate that TCP throughput decreases rapidly up to a point as the number of hops increases. It is argued that this decrease is due to the hidden terminals problem, which increases frames collisions. After a repeated transmission failure the MAC layer will react by two actions. First, the MAC will drop the head-of-line frame destined to the next hop we note that this type of drops is known also as drops due to contention on wireless channel. Second, the MAC will notify the upper layer about a link failure. When the routing protocol of a source node detects a routing failure, it will initiate a route re-establishment process. In general the route re-establishment duration is greater than the retransmission timer of the TCP agent; hence the TCP agent will enter the back procedure and will set its congestion window to 1. Also, as TCP sender's does not have indications on the route re-establishment event, TCP will sure from a long idle time. During this time, the network may be connected again, but TCP is still in the back state.

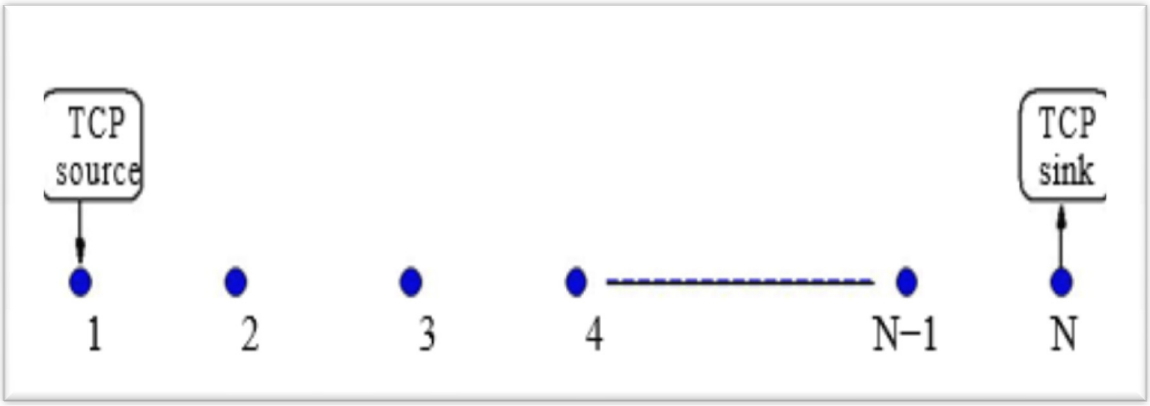


Figure 1. TCP static linear multi-hop chain

Proposals to improve TCP performance in ad hoc networks

We present the various proposals which have been made in the literature to improve the performance of TCP in ad hoc networks. We regroup these proposals to four sets according to the four problems. These four problems are:

1. TCP is unable to distinguish between losses due to route failures and those due to network congestion;
2. Frequent route failures,
3. Contention on wireless channel;
4. TCP unfairness.

We note that problems (1) and (2) are the main causes of TCP performance degradation in MANETs. However, problems (3) and (4) are the main causes of TCP performance degradation in SANETs. Figure 2 shows general classifications of each set of proposals. We classify the proposals that belong to the same set to two types: cross layer proposals, and layered proposals. The cross layer proposals rely on interactions between two layers of the Open System Interconnection (OSI) architecture. These proposals were motivated by the fact that providing lower layer information's to upper layer should help the upper layer to perform better. Thus, depending on between which two OSI layers there will be information's exchange, cross layer proposals can be further classed to four types: TCP and network, TCP and link, TCP and physical, and network and physical. Layered proposals rely on adapting OSI layers independently of other layers. Thus, depending on which layer is involved, layered proposals can be further classed to three types: TCP layer, network layer, and link layer proposals.

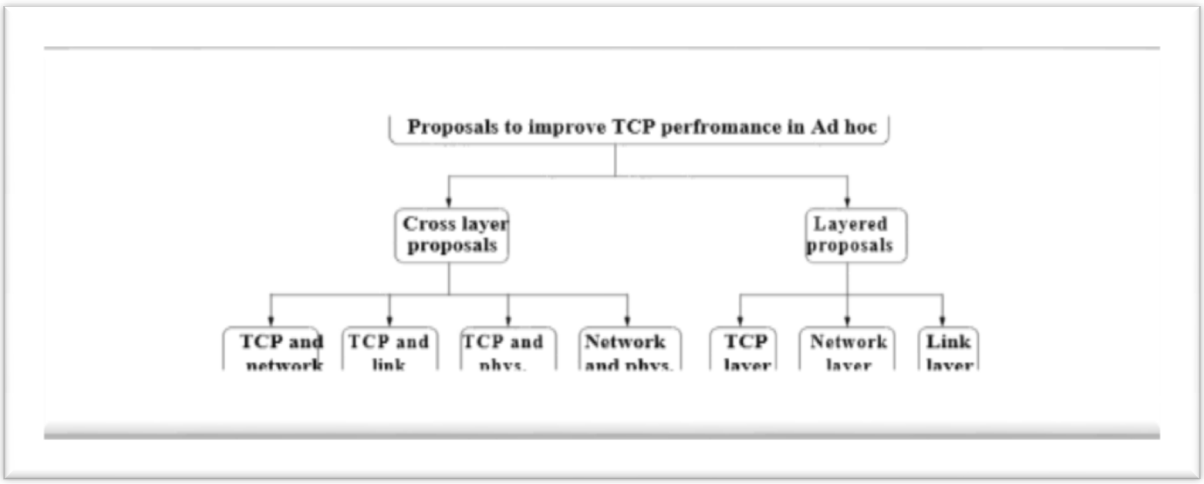


Figure 2. Classification of proposals to improve TCP performance in ad hoc networks

In general, cross layer solutions report better performance than layered solution. But layered solutions respect the concept of designing protocols in isolation, thus they are considered as long term solutions. So, to choose between cross layer and layered solutions we have to answer RST what is prior for us,

performance optimization or architecture Performance optimization can lead to short term gain, while architecture is usually based on longer term considerations. In addition, cross layer solutions are more complex to implement and to design than layered solutions. The reason is that the implementation of cross layer solutions requires at least medications of two OSI layers, and their design requires that the system is considered in its entirety.

TCP and network cross layer proposals

This metric **TCP-F: (**TCP Feedback) is a feedback based approach to handle route failures in MANETs. This approach allows the TCP sender to distinguish between losses due to routes failure and those due to network congestion. When the routing agent of a node detects the disruption of a route, it explicitly sends a Route Failure Notification (RFN) packet to the source. On receiving the RFN, the source goes into a snooze state. A TCP sender in snooze state will stop sending packets, and will freeze all its variables, such as timers and congestion window size. The TCP sender remains in this snooze state until it is notified of the restoration of the route through Route Re-establishment Notification (RRN) packet. On receiving the RRN, the TCP sender will leave the snooze state and will resume transmission based on the previous sender window and timeout values. To avoid blocking in the snooze state, the TCP sender, on receiving RFN, triggers a route failure timer. When this timer expires the congestion control algorithm is invoked normally. The simulation scenario is basic and is not based on an ad hoc network. Instead, they emulate the behavior of an ad hoc network from the viewpoint of a transport layer.

ELFN-based technique: Explicit Link Failure Notification technique is similar to TCP-F. However in contrast to TCP-F, the evaluation of the proposal is based on a real interaction between TCP and the routing protocol. This interaction aims to inform the TCP agent about route failures when they occur. The authors use an ELFN message, which is piggy-backed on the route failure message sent by the routing protocol to the sender. The ELFN message is like a host unreachable Internet Control Message Protocol (ICMP) message, which contains the sender receiver addresses and ports, as well as TCP packet's sequence number. On receiving the ELFN message, the source responds by disabling its retransmission timers and enters a standby mode. During the standby period, the TCP sender probes the network to check if the route is restored. If the acknowledgment of the probe packet is received, the TCP sender leaves the standby mode, resumes its retransmission timers, and continues the normal operations. The ELFN message is like a host unreachable Internet Control Message Protocol (ICMP) message, which contains the sender receiver addresses and ports, as well as TCP packet's sequence number. On receiving the ELFN message, the source responds by disabling its retransmission timers and enters a standby mode. During the standby period, the TCP sender probes the network to check if the route is restored. If the acknowledgment of the probe packet is received, the TCP sender leaves the standby mode, resumes its retransmission timers, and continues the normal operations.

In the mentioned reference, the authors study the act of varying the time interval between probe packets. Also, they evaluate the impact of the RTO and the Congestion Window (CW) upon restoration of the route. They end that a probe interval of 2 sec. performs the best, and they suggest making this interval a function of the RTT instead of giving it an axed value. For the RTO and CW values upon route restoration, they end that using the prior values before route failure performs better than initializing CW to 1 packet and/or RTO to 6 sec., the latter value being the initial default value of RTO in TCP Reno and New Reno versions. This technique provides sign cant enhancements over standard TCP, but further evaluations are still needed. For instance, die rent routing protocols should be considered other than the reactive protocol DSR considered, especially proactive protocols such as OLSR.

ATCP: Ad hoc TCP utilizes network layer feedback too. In addition to the route failures, ATCP tries to deal with the problem of high Bit Error Rate (BER). The TCP sender can be put into persist state, congestion control state or retransmit state. A layer called ATCP is inserted between the TCP and IP layers of the TCP source nodes. ATCP listens to the network state information provided by ECN (Explicit Congestion Notification) messages and by ICMP Destination Unreachable message; then ATCP puts TCP agent into the appropriate state. On receiving a Destination Unreachable message, TCP agent enters a persist state. The TCP agent during this state is frozen and no packets are sent until a new route is found by probing the network. The ECN is used as a mechanism to explicitly notify the sender about network congestion along the route being used. Upon reception of ECN, TCP congestion control is invoked normally without waiting for a timeout event. To detect packet losses due to channel errors, ATCP monitors the received ACKs [Aslanishvili, 2012]. When ATCP sees that three duplicate ACKs have been received, it does not forward the third duplicate ACK but puts TCP in the persist state and quickly retransmits the lost packet from TCP's buyer. After receiving the next ACK, ATCP will resume TCP to the normal state. Note that ATCP allows interoperability with TCP sources or destinations that do not implement ATCP.

Split TCP: TCP connections that have a large number of hops sure from frequent route failures due to mobility. To improve the throughput of these connections and to resolve the unfairness problem, the Split TCP scheme was introduced to split long TCP connections into shorter localized segments see Figure 3. The interfacing node between two localized segments is called a proxy. The routing agent decides if its node has the role of proxy according to the inter-proxy distance parameter. The proxy intercepts TCP packets, buyer them, acknowledges their receipt to the source (or previous proxy) by sending a local acknowledgment (LACK). Also, a proxy is responsible for delivering the packets, at an appropriate rate, to the next local segment. Upon the receipt of a LACK (from the next proxy or from the destination), a proxy will purge the packet from its buyer. To ensure the source to destination reliability, an ACK is sent by the destination to the source similarly to the standard TCP. In fact, this scheme splits also the transport layer functionalities into those end-to-end reliability and congestion control. This is done by using two transmission windows at the source which are the congestion window and the end-to-end window. The congestion window is a sub-window of the end-to-end window. While the congestion window changes in accordance with the rate of arrival of LACKs from the next proxy, the

end-to-end window will change in accordance with the rate of arrival of the end-to-end ACKs from the destination. At each proxy, there would be a congestion window that would govern the rate of sending between proxies.

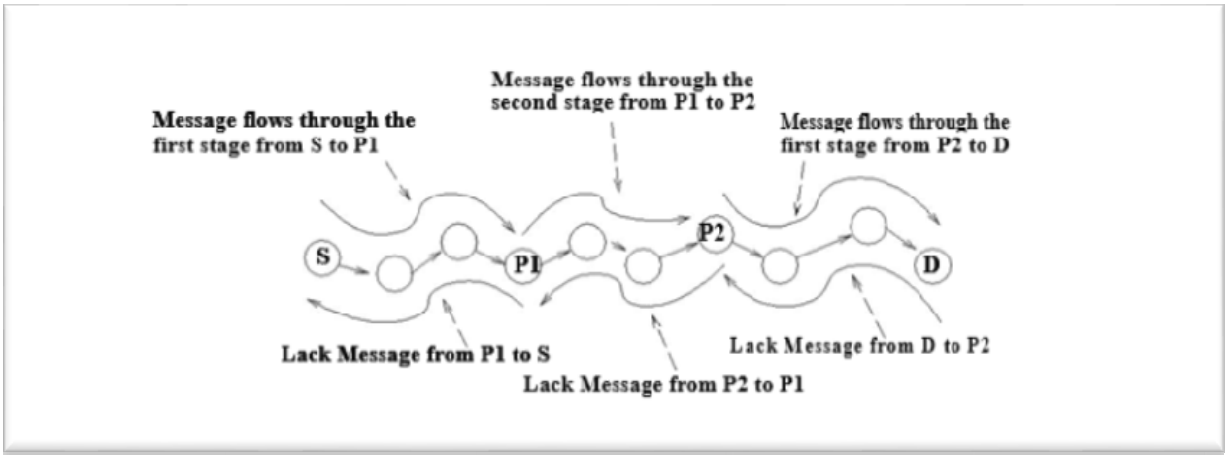


Figure 3. TCP connections are split into shorter localized segments

Simulation results indicate that an inter-proxy distance of between 3 and 5 has a good impact on both throughput and fairness. The authors report that an improvement up to 30% can be achieved in the total throughput by using Split TCP. Also, this proposal makes the role of proxy nodes more complex, as for each TCP session they have to control packet delivery to succeeding proxies.

Conclusion

We have presented the state-of-the-art of TCP over static and mobile ad hoc networks (SANETs and MANETs). The principal problem of TCP in this MANETs environment is clearly its inability to distinguish between losses induced by network congestion and other types of losses. [Aslanishvili, 2014] TCP assumes that losses are always due to network congestion. But while this assumption in most cases is valid in wired networks, it is not true in MANETs. In MANETs, there are indeed several types of losses, including losses cause by routing failures, by network partitions, and by high bit error rates. Performing congestion control in these cases, like TCP does, yields poor performance. In static multi-hop ad hoc networks the principal problem of TCP is the contention on wireless channel that induces routes failures and losses. In order to solve these problems, several proposals have been made in the literature. We classed these proposals as layered and cross layer proposals. In cross layer proposals, TCP and the underlying protocols cooperate to improve Ad hoc network performance. In layered proposals, one OSI layer is adapted. For example, TCP layer proposals require only the cooperation of the sender and receiver, like in TCP-DOOR and Fixed-RTO. However, cross layer proposals yield higher improvement than layered ones.

Bibliography

[Aslanishvili, 2012] Irma Aslanishvili, One model for two-hop relay Routing with limited Packet Lifetime the Conference for International Synergy in Energy, Environment, Tourism and contribution of Information Technology in Science, Economy, Society and Education era-7, 2012, ISSN 1791-1133, http://era.teipir.gr

[Aslanishvili, 2014] Irma Aslanishvili, Three RD Models for Two-Hop Relay Routing With Limited Packets Lifetime In Ad Hoc Networks", International Journal "Information Models and Analyses" Volume 3, Number 3, 2014

[Namicheishvili et al, 2011] Oleg Namicheishvili, Hamlet Meladze, Irma Aslanishvili, Transactions, "Two models for two-hop relay Routing with limited Packet Lifetime", Georgian Technical University, Automated Control Systems, 2011, No1(10), pp. 54-58.

Authors' Information

Irma Aslanishvili *– Iv.Javakhishvili Tbilisi State University, Faculty of Exact and Natural Sciences, teacher of Informatics and Computer Sensor Networks; e-mail: Irma.aslanishvili@tsu.ge*

Scientific Research: General theoretical information research, information systems and computer sensor networks

Tariel Khvedelidze*-Iv.Javakhishvili Tbilisi State University, Faculty of Exact and Natural Sciences, professor of Computer Sciences e-mail: tariel.khvedelidze@tsu.ge*

Scientific Research: General theoretical information research, information systems and computer Sciences

METHOD OF ESTIMATING RELIABILITY OF INFORMATION TRANSMISSION IN WIRELESS NETWORKS CHANNELS INCREASE IN NOISE AND INTERFERENCE

Sergey Zaitsev

Abstract: *This paper describes a method for estimating the reliability of information transmission in wireless networks operating in conditions of high noise and interference. The method is based on obtaining analytical relationships for calculating the reliability of information transmission based on nonlinear regression analysis and the use of log-likelihood function about transmitted bits in decision-making in the turbo decoder.*

Keywords: information technologies, a priori uncertainty, wireless networks, turbo codes

ACM Classification Keywords: *H.4 Information system applications, H.4.3 Communications Applications and Expert Systems, G.1.6 Optimization, I.6 Simulation and Modeling, K.6.4 System Management*

Introduction

To increase the reliability of information transmission in the current wireless systems use codes noise-resistant: block codes, BCH codes, Reed-Solomon Reed-Muller codes concatenated convolutional codes, turbo codes (TC) and others. The most effective among these are TC, which are energy efficiency inferior to the theoretical Shannon limit is only 0.5 dB [Berrou, 1993].

Today, due to the increased energy efficiency of TC used in mobile communication systems of the third generation 3G (cdma2000, cdma2000 1xEV-DO, cdma2000 1xEV-DV, UMTS), the system LTE, CCSDS in communication systems for transmission of telemetry data from spacecraft, systems, satellite digital TV DVB-RCS [Holma, 2006- Ergen, 2009] and others.

A significant impact on reducing the reliability of information transmission in wireless networks have a powerful noise and intentional interference from, including jamming devices. To increase the reliability of information transmission on the physical layer of wireless systems today use adaptive-code signal structure using a signal multiposition and turbo codes. Development and implementation of adaptive methods for information exchange requires the establishment of effective procedures for monitoring and forecasting of states of channels and reliability of information transmission.

Analysis of research and publications

The basic methods and algorithms for the estimation of reliability of information transmitted over wireless networks presented in [Fengqin, 2003- Roshanzadeh, 2012].

In [Fengqin, 2003] to evaluate the reliability of the information used by the values of log-likelihood function of the transmitted bits in association with a dispersion of noise in the channel with additive white Gaussian noise. Calculated the likelihood function of the transmitted bits:

$$L(x_t) = \log_e \frac{\Pr\{x_t = +1 \mid z_t\}}{\Pr\{x_t = -1 \mid z_t\}} = \log_e \frac{f(z_t \mid x_t = +1)}{f(z_t \mid x_t = -1)} = \frac{2}{\sigma_m^2} z_t , \quad (1)$$

where x_t – the transmitted bits, z_t – accepted, $f(\cdot \mid \cdot)$ – probability density function, $t \in \overline{1, N}$, σ_m^2 – noise variance.

Is the mean value (1):

$$E[|L(x)|] = \frac{2}{\sigma_m^2} E[|z|], \quad (2)$$

$$E[|z|] = \sqrt{\frac{2}{\pi}} \sigma_m e^{-1/(2\sigma_m^2)} + erf\left(\frac{1}{\sqrt{2\sigma_m^2}}\right), \quad (3)$$

where $erf(\cdot)$ – the error function.

If $\sigma_m^2 < 0.2$, $E[|z|] \approx 1$, and therefore

$$\sigma_m^2 \approx \frac{2}{E[|L(x)|]}, \text{ when } E[|L(x)|] > 10 . \quad (4)$$

Evaluating the results presented it can be concluded that this evaluation method is suitable for small values of noise variance, i.e. for large signal-to-noise ratio.

In addition, the authors of [Fengqin, 2003] proposed to evaluate the use of error codes CRC, which are described by polynomials $D^{12} + D^{11} + D^3 + D^2 + 1$, $D^{16} + D^{15} + D^2 + 1$, $D^{16} + D^{12} + D^5 + 1$. Study the properties of CRC-codes are shown in [Castagnoli, 1990].

It should be noted that the CRC-codes have a low ability to detect errors that the large noise variance reduces the number of detected errors, and reduces the reliability of the transmitted information.

Studies on the use Reed-Solomon codes to detect errors in the received sequence shown in [Fadnavis, 2013].

Using additional error detection codes leads to the introduction of redundancy into the transmitted sequence, and to process complexity in a digital signal processing means to transmit information, which leads to an excessive use of computing resources systems.

Further analysis of [Fengqin, 2003- Roshanzadeh, 2012] leads to the conclusion that many modern methods for assessing the reliability based on the calculation of noise variance and the use of error detection codes. In the presence of strong noise in the channel, existing approaches will result in lower quality evaluation reliability of information transmission.

Therefore, there is a need to develop new, more efficient methods for assessing the reliability of the transmitted information channels for wireless data transmission systems with high noise levels.

Description transmission system

Figure 1 shows a block diagram of a data transmission system that operates at high noise and interference.

Data source generates a sequence $\overline{U}$ that enters the encoder TC. TC encoder produces a sequence of encoded bits $\overline{Y}$. In modulator, BPSK sequence $\overline{Y}$ converted into modulated signals $\overline{S}$. Frequency tuning is performed using a frequency synthesizer (FS) and a generator of pseudorandom sequence hopping (GPRS) in the pseudorandom modulator operating frequency (PRMOF). In the communication channel signals affect the fluctuation noise and jamming, the most commonly used noise barrage interference, noise interference in the part of the band polyharmonic interference and response interference. The received signal output from the receiver is supplied to a demodulator PRMOF, where due to the frequency synthesizer controlled by GPRS, the operating frequency jumps are eliminated. Character sequence after the BPSK demodulation and iterative decoding in the decoder TC is converted to a bit sequence received $\overline{U}'$. Furthermore, the information supplied to the decoding device reliability analysis.

In [Holma, 2006] is a block diagram of the encoder and iterative decoder TC with "soft" input and "soft" output.

Using a decoder with a "soft" input and "soft" output indicates that the data input and output of the decoder TC are of type real numbers, which significantly improves the quality of decoding [Berrou, 1993], in contrast to the "hard" option when using two-level decision making data and take only two integers "0" or "1".

TC encoder consists of two parallel recursive systematic convolutional codes (RSCC), separated pseudorandom interleaver. An interleaver in the turbo code encoder composition is used for shifting the information bits input to the second RSCC, to reduce the correlation of the first parity bit and second RSCC. Each RSCC performs encoding on an information sequence of its trellis diagram, the structure of which depends on the generator polynomial RSCC [Valenti, 2001]. TC iterative decoder consists of series-connected component decoders. One decoding iteration and decoder comprises two pseudorandom interleaving two devices (deinterleaving). Deinterleaving apparatus deinterleaving

operation is performed. In the decoder circuit TC used $d = 2I$ component decoders, wherein *I* – the total number of iterations of the decoding, $d \in \overline{2, D}$, *D* – the total number of component decoders.

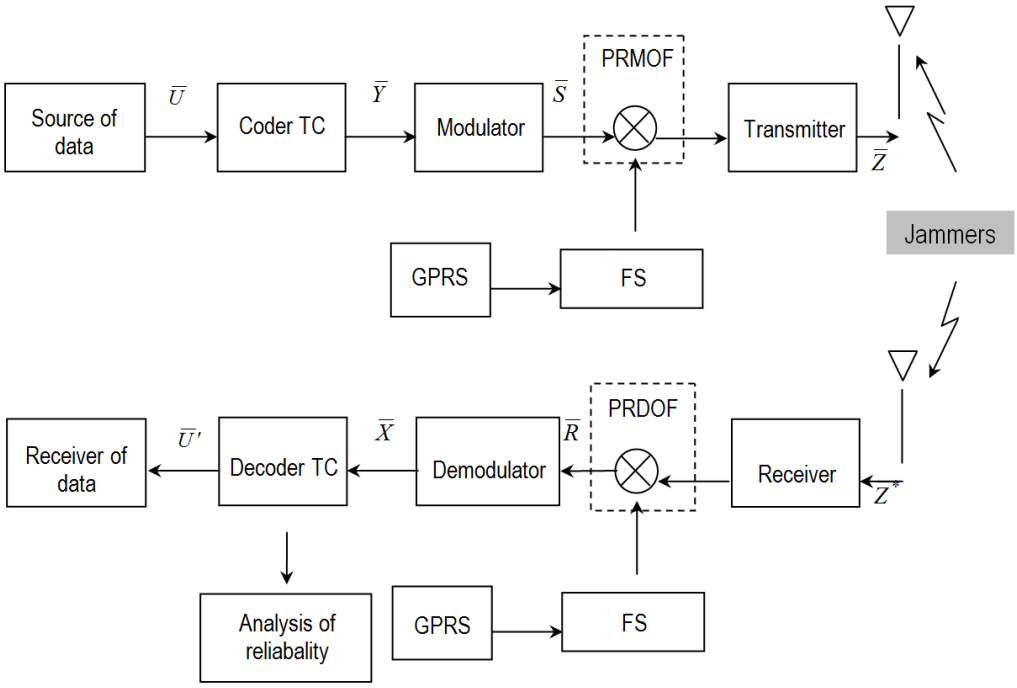


Figure 1.

Decoding of bits

Consider in more detail the operation of the decoder 1, 2 at the *j*-th decoding iteration decoding *t*-th bit, $j \in \overline{1, I}$, $t \in \overline{1, N}$, *N* – total number of transmitted bits in a data block. Let x_t – is *t*-th transmitted bit, and y_t – received *t*-th bit distorted by the influence of white Gaussian noise. As in channels with high noise levels at the receiver decisions under conditions of uncertainty, the "soft" decision or logarithmic ratio of the likelihood function (LRLF), distance decoder 1 in the *j*-th iteration is determined by the following expression [Valenti, 2001]:

$$L^{1,j}(x_t \mid y_t) = \ln \frac{P(y_t \mid x_t = +1)}{P(y_t \mid x_t = -1)} + \ln \frac{P(x_t = +1)}{P(x_t = -1)} = L_a^{1,j}(x_t) + L^{1,j}(y_t \mid x_t), \quad (5)$$

where $L^{1,j}(y_t \mid x_t)$ – LRLF, which is obtained by measuring y_t outlet channel interleave conditions that can be transmitted $x_t = +1$ or $x_t = -1$, a $L_a^{1,j}(x_t)$ – priori LRLF data bit x_t.

To simplify equation (1) can be rewritten as follows:

$$L^{1,j}(x_t) = L_c^{1,j}(y_t) + L_a^{1,j}(x_t) + L_e^{1,j}(x_t), \quad (6)$$

where $L_c^{1,j}(y_t)$ – the "channel reliability", $L_e^{1,i}(x_t)$ – posteriori LRLF data bit x_t.

The latter is calculated a posteriori LRLF data bit x_t – $L_e^{1,j}(x_t)$:

$$L_e^{1,j}(x_t) = L^{1,j}(x_t) - L_c^{1,j}(y_t) - L_a^{1,j}(x_t). \quad (7)$$

Interleaver converts posteriori LRLF $L_e^{1,j}(x_t)$ a priori LRLF $L_a^{2,j}(x_t)$: $L_a^{2,j}(x_t) = f_1\left(L_e^{1,j}(x_t)\right)$, which is supplied to the decoder 2. The decoder 2 performs the following calculation to obtain the posterior LRLF data bit x_t – $L_e^{2,j}(x_t)$:

$$L_e^{2,j}(x_t) = L^{2,j}(x_t) - L_c^{2,j}(y_t) - L_a^{2,j}(x_t). \quad (8)$$

After surgery, deinterleaving $L_a^{1,j+1}(x_t) = f_2\left(L_e^{2,j}(x_t)\right)$ value $L_a^{1,j+1}(x_t)$ is used as a priori for one iteration of the decoder. Further evaluation is carried out similar to (7) and (8).

After performing the required number of iterations, or in the case of a forced stop the iterative decoding procedure, submitted the "hard" decision on the decoded bits:

$$\widetilde{x}_t^{\mathrm{C}} = \begin{cases} 1, \ if \ L(x_t^{\mathrm{C}}) \geq 0 \\ 0, \ if \ L(x_t^{\mathrm{C}}) < 0 \end{cases}. \quad (9)$$

Find of error area

To assess the quality of the decoding will use the following algorithm.

1. Formation of a matrix of values of systematic data bits X^{S} size $1 \times N$, produced by the encoder turbo code:

$$X^{\mathrm{S}} = \left[x_1^{\mathrm{S}} \quad x_2^{\mathrm{S}} \quad \ldots \quad x_N^{\mathrm{S}}\right]. \quad (10)$$

2. Formation matrix *LA* size values of a priori information for the *i*-th decoder:

$$LA = \left[L_a^i(x_1^{\mathrm{S}}) \quad L_a^i(x_2^{\mathrm{S}}) \quad \ldots \quad L_a^i(x_N^{\mathrm{S}})\right]. \quad (11)$$

3. Formation of systematic information channel matrix samples Y^{S} size $1 \times N$:

$$Y^{\mathrm{S}} = \left[y_1^{\mathrm{S}} \quad y_2^{\mathrm{S}} \quad \ldots \quad y_N^{\mathrm{S}}\right]. \quad (12)$$

4. Formation of the channel matrix of test samples $Y^{\mathrm{P}j}$ size $1 \times N$, $j \in \overline{1,K}$, where K – number of encoders (decoders turbo code):

$$Y^{Pj} = \left[y_1^{\mathrm{P}j} \quad y_2^{\mathrm{P}j} \quad \ldots \quad y_N^{\mathrm{P}j}\right]. \quad (13)$$

5. Formation of the matrix size *LE* values "external" information for the *i*-th decoder.

$$LE = \left[L_e^i(x_1^{\mathrm{C}}) \quad L_e^i(x_2^{\mathrm{C}}) \quad \ldots \quad L_e^i(x_N^{\mathrm{C}})\right]. \quad (14)$$

6. Calculating $L^i(x_t^{\mathrm{C}})$, $i \in \overline{1,D}$, $t \in \overline{1,N}$ for the *i*-th decoder and bit block of *N* according to expression (6).

7. Formation matrix *L* size $1 \times N$ LRLF for the i-th decoder.

$$L = \left[L^i(x_1^{\mathrm{C}}) \quad L^i(x_2^{\mathrm{C}}) \quad \ldots \quad L^i(x_N^{\mathrm{C}}) \right]. \tag{15}$$

8. Formation matrix *LE* size for the *i*-th decoder.

$$L^* = \left[L^i(x_1^{\mathrm{C}}) x_1^{\mathrm{C}} \quad L^i(x_2^{\mathrm{C}}) x_2^{\mathrm{C}} \quad \ldots \quad L^i(x_N^{\mathrm{C}}) x_N^{\mathrm{C}} \right]. \tag{16}$$

9. The loop: if $L^i(x_t^{\mathrm{C}}) x_t^{\mathrm{C}} < 0$, $t \in \overline{1,N}$, then $S_L = S_L + 1$.

Simulation was performed in the Borland C ++ Builder data transmission system using a turbo code with a two generator polynomial (1, 7/5) recursive systematic convolutional code, a pseudo-random interleaving algorithm, the decoding algorithm Map, no perforation 1, 2, 4 and 8 decoding iterations, the number of bits transmitted in a bit unit, for various values of the ratio of signal energy to noise power spectral density (signal-to-noise ratio - SNR). The simulation results were processed using a package of Matlab.

After tests, we obtain a distribution of errors, that is, the dependence of the normalized mean value of the number of errors $S_L^* = \frac{S_L}{Nn}$ the logarithmic likelihood ratio $L(x^{\mathrm{C}})$, as shown in Figures 2-5, where *N* – number of bits in a transmitted block, *n* - the number of blocks of size *N*.

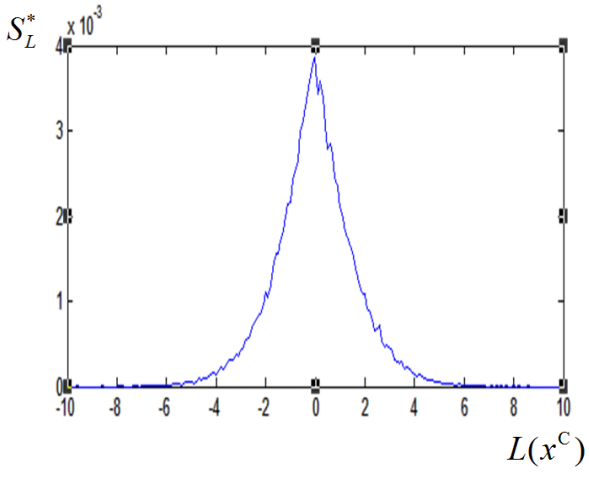


Figure 2 - Distribution of values of error (Iteration - 1, SNR – 0.3 dB)

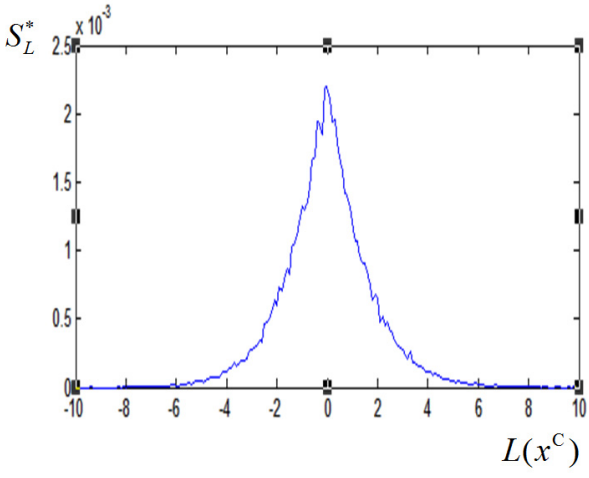


Figure 3 - Distribution of values of error (Iteration - 2, SNR – 0.3 dB

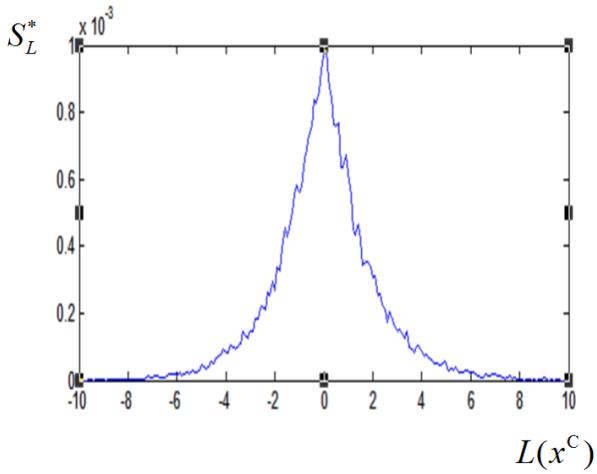


Figure 4 - Distribution of values of error (Iteration - 4, SNR - 0.3 dB)

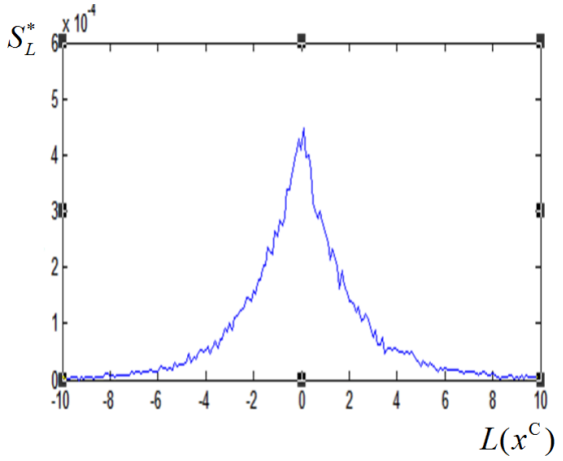


Figure 5 - Distribution of values of error (Iteration - 8, SNR - 0.3 dB)

As you can see, the distribution of the error values shown in Figure 1-4, describes a normal (Gaussian) distribution law of random variables

$$f(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{17}$$

where μ – mathematical delay, σ – the standard deviation.

To find the area of the field error at a given value SNR need to find the integral of the function, which describes the distribution of values, and use the rule of three sigma. This will get 99.7% of the values, so we can say that will be evaluated for 99.7% of the area (Figure 6).

$$F(x)=\int_{-(3\sigma-\mu)}^{3\sigma+\mu}\frac{1}{\Delta\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{18}$$

where Δ – range separating the abscissa with the graphical display of the distribution function on the graph.

On the basis of of the experiments, we assume that μ=0 and because the formula (18) is simplified:

$$F(x)=\int_{-3\sigma}^{3\sigma}\frac{1}{\Delta\sigma\sqrt{2\pi}}e^{-\frac{x^2}{2\sigma^2}}. \tag{19}$$

Let $a=\frac{1}{\sigma\sqrt{2\pi}}$, $c=\sqrt{2}\sigma$, then

$$f(x) = ae^{-\frac{(x-b)^2}{c^2}}, \tag{20}$$

$$F(x) = \int_{-3\sigma}^{3\sigma} \frac{a}{\Delta} e^{-\left(\frac{x-b}{c}\right)^2}. \tag{21}$$

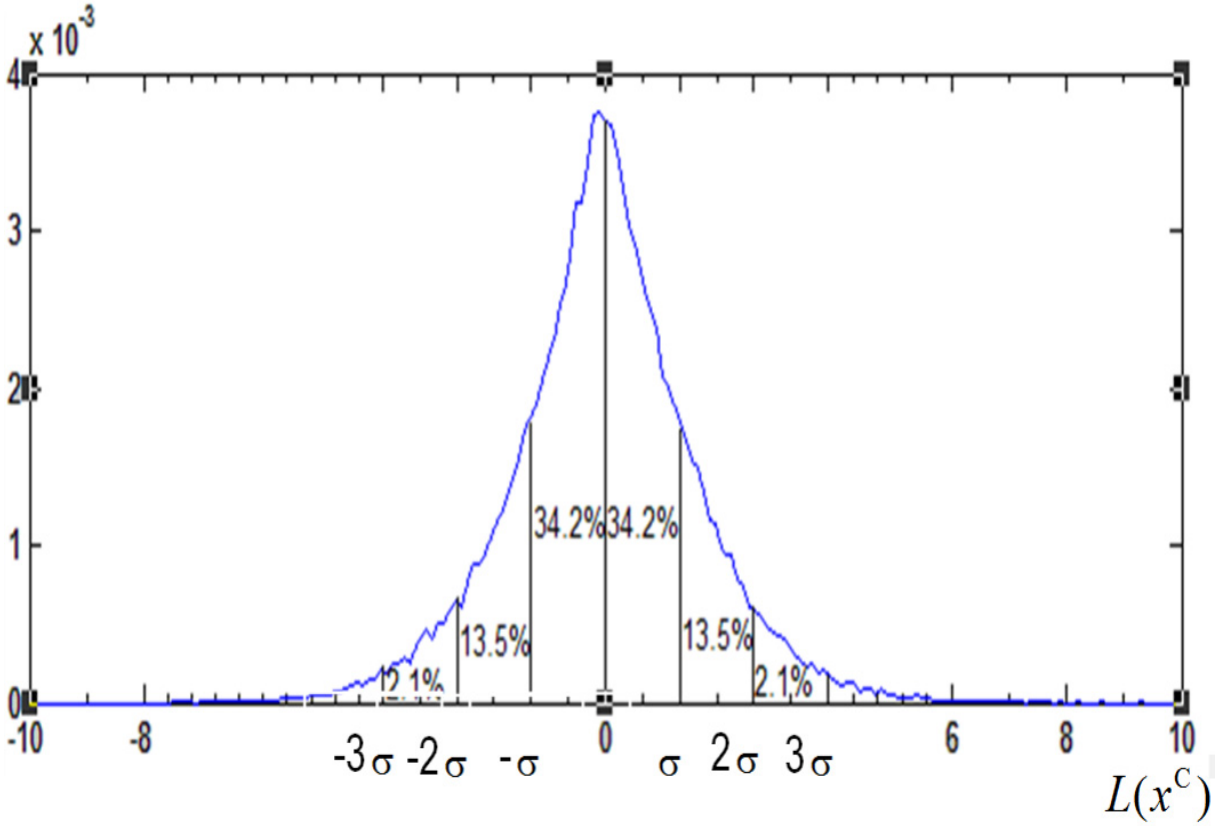


Figure 6 - Percentage of hitting errors in the range of values $L(x^C)$ (Iteration - 1, SNR - 0.3 dB)

Regression analysis

Logarithm of both sides of equation (20), assuming that b = 0:

$$\ln(f(x)) = \ln\left(ae^{-\frac{(x-b)^2}{c^2}}\right) = \ln(a) - \frac{x^2}{c^2}. \tag{22}$$

We introduce new variables and parameters: $b_0 = \ln(a)$, $b_1 = -\frac{1}{c^2}$, $Y = \ln(f(x))$, $X = x^2$.

Obtain the regression model, specified in the linear form:

$$Y = b_0 + b_1 X. \tag{23}$$

We find the parameters of the model using the least squares method. To do this, we write the system of normal equations for the linear model:

$$\begin{cases} nb_0 + b_1 \sum_{i=1}^{n} X_i = \sum_{i=1}^{n} Y_i \\ b_0 \sum_{i=1}^{n} X_i + b_1 \sum_{i=1}^{n} X_i^2 = \sum_{i=1}^{n} (X_i Y_i) \end{cases}, \quad (24)$$

where n - number of observations. System parameters are calculated by the formulas:

$$b_0 = \frac{\sum Y_i}{n} - b_1 \frac{\sum X_i}{n}, \quad (25)$$

$$b_1 = \frac{n\sum (Y_i X_i) - \sum X_i \sum Y_i}{n\sum X_i^2 - (\sum X_i)^2}, \quad (26)$$

To find the coefficients of the variables must be expressed: $a = e^{b_0}$, $c^2 = -\frac{1}{b_1}$.

For factors with which we will continue to determine the area of the field errors, we carry out a regression analysis using the data presented in Figures 7-10.

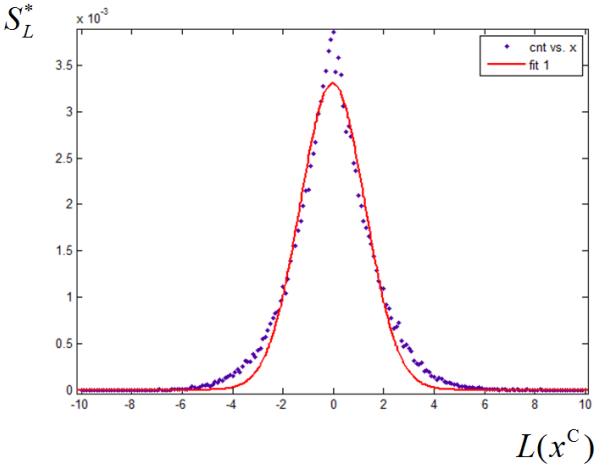


Figure 7 - Regression analysis for field errors (Iteration - 1, SNR - 0.3 dB)

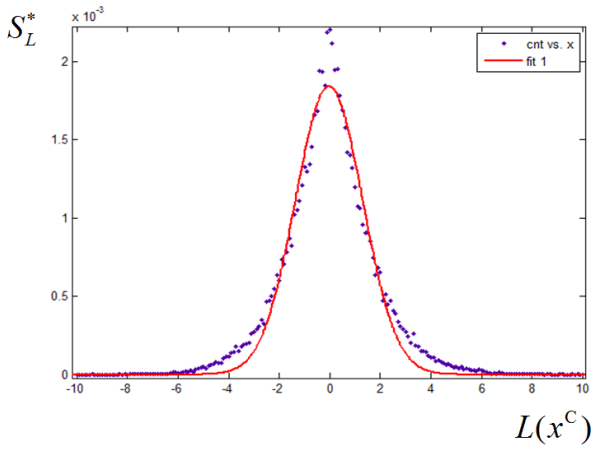


Figure 8 - Regression analysis for field errors (Iteration - 2, SNR - 0.3 dB)

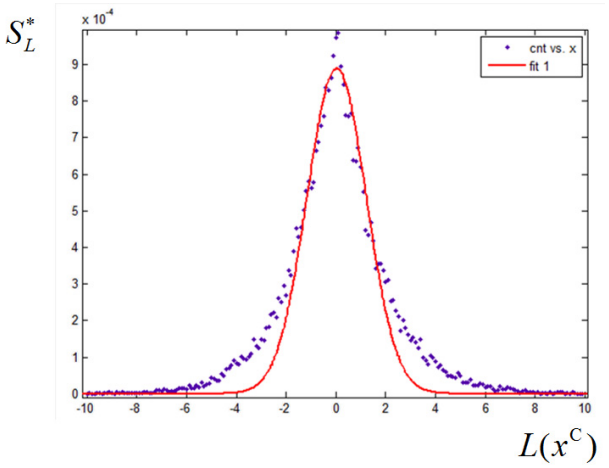


Figure 9 - Regression analysis for field errors (Iteration - 4, SNR - 0.3 dB)

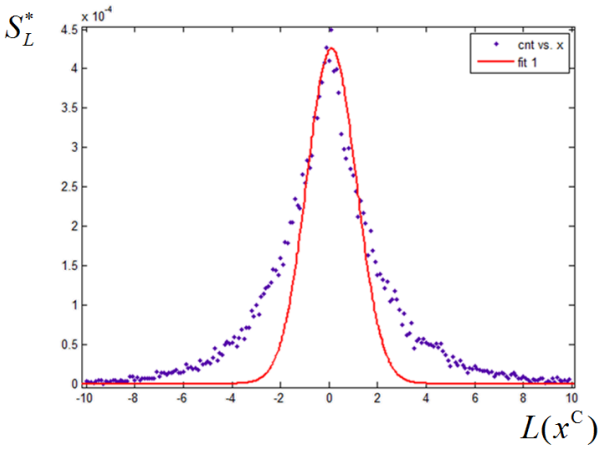


Figure 10 - Regression analysis for field errors (Iteration - 8, SNR - 0.3 dB)

Table 1 shows the results of finding the coefficients of the functions of normal error distributions a, b, c.

Table 1 - Results of finding the coefficients of the functions of a normal distribution of errors

SNR, дБ	Iterations			
	1	2	4	8
0	a = 0,004062, b = – 0,01998, c = 1,753	a = 0,002824, b = – 0,03997, c = 1,845	a = 0,002001, b = – 0.03508, c = 1,888	a = 0,001571, b = – 0,04475, c = 1,868
0,03	a = 0,003982, b = – 0,02925, c = 1,749	a = 0,002724, b = – 0.02381, c = 1,845	a = 0,001823, b = – 0.03427, c = 1,906	a = 0,001418, b = – 0,03458, c = 1,881
0,06	a = 0,003836, b = – 0,02161, c = 1,775	a = 0,002563, b = – 0,03255, c = 1,882	a = 0,001755, b = – 0.02994, c = 1,91	a = 0,001166, b = – 0,03461, c = 1,953
0,09	a = 0,003851, b = – 0,02754, c = 1,761	a = 0,002498, b = – 0,02888, c = 1,837	a = 0,001621, b = – 0,01116, c = 1,832	a = 0,001146, b = – 0,008424, c = 1,927

The coefficients obtained with a 95% confidence interval, the sum of squared errors SSE: 4,264e006, the value of determination R-square: 0,9873, the customized value of determination Adjusted R-square: 0,9873, the standard deviation RMSE: 8,452e-005. With the value of determination and rigged values of determination can say about the effectiveness and reliability of the selected coefficients of the selected function regression.

Let *L* - random variable whose values are the results of decoding the *i*-th decoder, namely the calculation of the transferred LRLF in an *n*-bit block length *N*: $L^i\left(x_{kt}^C\right)$, $t \in \overline{1,N}$, $k \in \overline{1,n}$. Expectation and variance of the random variable *L* is defined by the following expressions:

$$M_L = \frac{\sum_{k=1}^{n}\sum_{t=1}^{N} L^i\left(x_{kt}^C\right)}{nN}, \tag{27}$$

$$D_L = \frac{\sum_{k=1}^{n}\sum_{t=1}^{N}\left(L^i\left(x_{kt}^C\right) - M_L\right)^2}{(n-1)(N-1)}. \tag{28}$$

Thus, by analyzing the variance of transmitted bits LRLF can compute the coefficients of the normal distribution function to calculate the error and the accuracy of information transmission. The number of blocks n to calculate the expectation and variance calculated from the required values and a confidence level of the confidence interval [Иващенко, 1988].

Conclusion

This paper presents a method for estimating the reliability of information transmission in wireless systems that operate in channels with elevated levels of noise and interference. The method is based on the results of the decoding of turbo codes to determine the value of reliability of information transmission. The results of the simulation method for assessing the reliability of the transmitted information through statistical estimation of log-likelihood function of the transmitted bits, which are produced in an iterative turbo decoder code. The results make it possible to estimate the variance LRLF about bits transmitted and received using the coefficients to calculate the values of reliability of information transmission in the wireless channel. The results can be applied in the design and development of adaptive data transmission systems with given values of reliability.

Bibliography

[Berrou, 1993] Berrou C. Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes / C. Berrou, A. Glavieux, P. Thitimajshima // Proc. Int. Conf. On Commun., May 1993. – 1993. – P. 1064 - 1070.

[Holma, 2006] Holma H. HSDPA/HSUPA for UMTS: High Speed Radio Access for Mobile Communications / H. Holma, A. Toskala. – John Wiley & Sons, 2006. – 268 p.

[Valenti, 2001] Valenti M. The UMTS turbo code and an efficient decoder implementation suitable for software-defined radios /M. Valenti, J. Sun // Int. Journal of Wireless Inf. Networks. – 2001. – Vol. 8, №4. – P. 203-215.

[Blue Book, 1998] Consultative Committee for Space Data Systems “Recommendations for space data systems, telemetry channel coding” // BLUE BOOK. – 1998. – May.

[3GPP, 2005] 3GPP TS 25.212. Multiplexing and Channel Coding (FDD). 3GPP Technical Specification Group Radio Access Network, March 2005.

[DVB, 2000] Digital Video Broadcasting (DVB): Interaction channel for satellite distribution systems (DVB-RCS) // ETSI EN 301 790 V1.2. 1 (2000-07). – 2000. – February.

[Ergen, 2009] Ergen M. Mobile Broadband. Including WiMax and LTE / M. Ergen. – Springer, 2009. – 513 p.

[Fengqin, 2003] Fengqin Zhai. Techniques for Early Stopping and Error Detection in Turbo Decoding / Fengqin Zhai, Ivan J. Fair // IEEE Transactions on Communications. – 2003. – Vol. 51, No. 10. – P. 1617-1623.

[Castagnoli, 1990] Castagnoli G. Optimum cyclic redundancycheck codes with 16-bit redundancy / G. Castagnoli, J. Ganz, P. Graber // IEEE Transactions on Communications. – 1990. – Vol. 38. – P. 111-114.

[Fadnavis, 2013] Fadnavis Shubham. Performance Optimization of HVD: An Error Detection and Correction Code / Fadnavis Shubham // Research Journal of Engineering Sciences. – 2013. – Vol. 2 (6). – P. 25-29.

[Yibing, 2012] Yibing Li. Analysis of BLER Performance for LTE Uplink Baseband Simulation System / Yibing Li, Xin Gui, Fang Ye // Journal of Computational Information Systems. – 2012. – Vol. 7. – P. 2691-2699.

[Agraval, 2012] Agraval V. Review of Reed Solomon Code for Error Detection and Correction / Agraval V., Pankaj G., Gaurav M. // International Journal of Research in IT, Management and Engineering. – 2012. – Vol. 2. – P. 105-113.

[Yuejun, 2013] Yuejun Wei. A CRC-Aided Hybrid Decoding Algorithm for Turbo Codes / Yuejun Wei., Ming Jiang, Bin Xia, Wen Chen, Yubang Yang // IEEE Wireless Communications Letters. – 2013. – Vol. 2. – P. 471-474.

[Egwali, 2013] Egwali A. Performance Evaluation of AN-VE: An Error Detection and Correction Code / Egwali A., Akwukwuma V. // African Journal of Computing & ICT. – 2013. – Vol. 6, No 1. – P. 117-126.

[Roshanzadeh, 2012] Roshanzadeh M. Error Detection & Correction in Wireless Sensor Networks by Using Residue Number Systems / Roshanzadeh M., Saqaeeyan S. // I.J. Computer Network and Security. – 2012. – 2. – P. 29-35.

[Иващенко, 1988] Иващенко А. В. Основы моделирования сложных систем на ЭВМ / А. В. Иващенко, Р. П. Сыпченко. – ЛВВИУС, 1988. – 272 с.

Authors' Information

Sergey Zaitsev *– Ph.D, Chernihiv National Technological University, 95, Shevchenko street, Chernihiv-27, Ukraine, 14027; e-mail: zas79@mail.ru*

Major Fields of Scientific Research: modeling of communications systems, adaptation of the wireless networks, turbo codes.

TABLE OF CONTENTS